CLC **Main** Workbench

USER MANUAL

Manual for
CLC Main Workbench 7.0
Windows, Mac OS X and Linux

February 19, 2014

This software is for research purposes only.

Contents

27 Expression analysis

28 Cloning and cutting

29 Sequence alignment

30 Phylogenetic trees

Part I

Introduction

Chapter 1

Introduction to *CLC Main Workbench*

Contents

Welcome to *CLC Main 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 Main Workbench* is developed by:

CLC bio,a QIAGEN Company
Silkeborgvej 2
Prismet
8000 Aarhus C
Denmark

http://www.clcbio.com

VAT no.: DK 28 30 50 87

Telephone: 45 70 22 32 44
Fax: +45 86 20 12 22

E-mail: info@clcbio.com

If you have questions or comments regarding the program, you are welcome to contact our support team. The best way to do this is from within the Workbench, using the menu option:

> **Help | Contact Support**

The Support team can also be reached by email at:

support@clcbio.com

When contacting Support by email, please include your license information. Information about how to to find your license information is included in the licenses section of our Frequently Asked Questions (FAQ) area: http://www.clcbio.com/faq

1.2 Download and installation

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

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]

1.2.2 Installation on Microsoft Windows

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

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

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 Main Workbench* and click **Next**.

- Choose if *CLC Main Workbench* should be used to open CLC files and click **Next**.

- Choose where you would like to create shortcuts for launching *CLC Main Workbench* and click **Next**.

- Choose if you would like to associate .clc files to *CLC Main Workbench*. If you check this option, double-clicking a file with a "clc" extension will open the *CLC Main Workbench*.

- Wait for the installation process to complete, choose whether you would like to launch *CLC Main 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 the following way:

When you have downloaded an installer:
      Locate the downloaded installer and double-click the icon.
      The default location for downloaded files is your desktop.
      Launch the installer by double-clicking on the "*CLC Main 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**.

---

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

- Choose where you would like to install the application and click **Next**.

- Choose if *CLC Main Workbench* should be used to open CLC files and click **Next**.

- Choose whether you would like to create desktop icon for launching *CLC Main Workbench* and click **Next**.

- Choose if you would like to associate .clc files to *CLC Main Workbench*. If you check this option, double-clicking a file with a "clc" extension will open the *CLC Main Workbench*.

- Wait for the installation process to complete, choose whether you would like to launch *CLC Main 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 CLCMainWorkbench_7_JRE.sh
```

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:

```
# clcmainwb7
```

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

```
# ./clcmainwb7
```

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 CLCMainWorkbench_7_JRE.rpm
```

Installation of RPM-packages usually requires root-privileges.

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

```
# clcmainwb7
```

1.3   System requirements

The system requirements of *CLC Main Workbench* are these:

- Windows XP, Windows Vista, Windows 7, Windows 8, Windows Server 2003 or Windows Server 2008

- Mac OS X 10.6 or later. However, Mac OS X 10.5.8 is supported on 64-bit Intel systems.

- Linux: RHEL 5.0 or later. SUSE 10.2 or later. Fedora 6 or later.

- 32 or 64 bit

- 1 GB RAM required

- 2 GB RAM recommended

- 1024 x 768 display required

- 1600 x 1200 display recommended

Special requirements for the 3D Molecule Viewer

- **System requirements**

    - A graphics card capable of supporting OpenGL 2.0.
    - Updated graphics drivers. Please make sure the latest driver for the graphics card is installed.

- **System Recommendations**

    - A discrete graphics card from either Nvidia or AMD/ATI. Modern integrated graphics cards (such as the Intel HD Graphics series) may also be used, but these are usually slower than the discrete cards.
    - A 64-bit workbench version is recommended for working with large complexes.

1.4 Licenses

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

To install a license, you must be running the program in administrative mode [2].
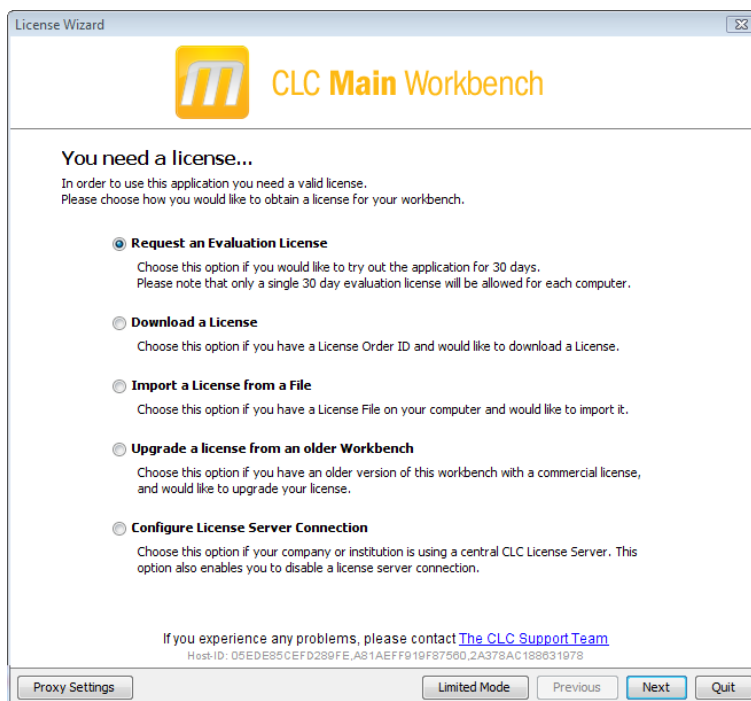
Figure 1.1: *The license assistant showing you the options for getting started.*

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 get a license ID. This is used here to download a license assocaited with 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 Main Workbench*, and you are entitled to upgrading to the new *CLC Main Workbench 7.0*, 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**.

To use the Download option in the License Manager, your machine must be able to access the external network. If this is not the case, please see the section 1.4.6.

---

[2]"How to do this differs for different operating systems. To run the program in administrator mode on Windows Vista, or 7, right-click the program shortcut and choose "Run as Administrator."

If for some reason you don't have a license ID or access to a license, you can click the **Limited Mode** button (see section 1.4.7).

1.4.1 Request an evaluation license

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

Each user is entitled to 30 days demo of *CLC Main Workbench*. If you need more time for evaluating, another two weeks of demo can be requested.

When you select to request an evaluation license, you will see the dialog shown in figure 1.2.
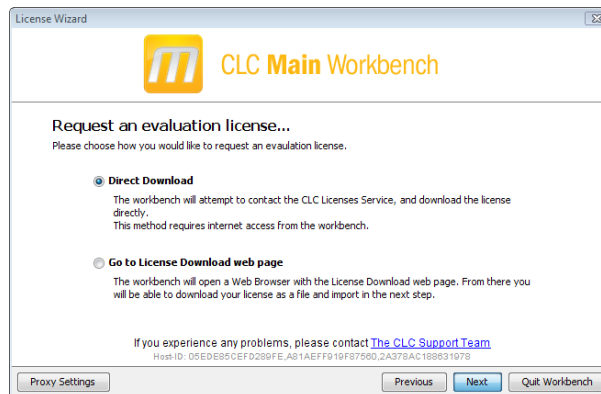
Figure 1.2: *Choosing between direct download or download web page.*

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.

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.

Figure 1.3: *A license has been downloaded.*

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.

Figure 1.5: *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 1.6.

Figure 1.6: *Read the license agreement carefully.*

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

1.4.2 Download a license

When you purchase a license, you get a license ID from CLC bio. This is used here to download a license associated with 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).
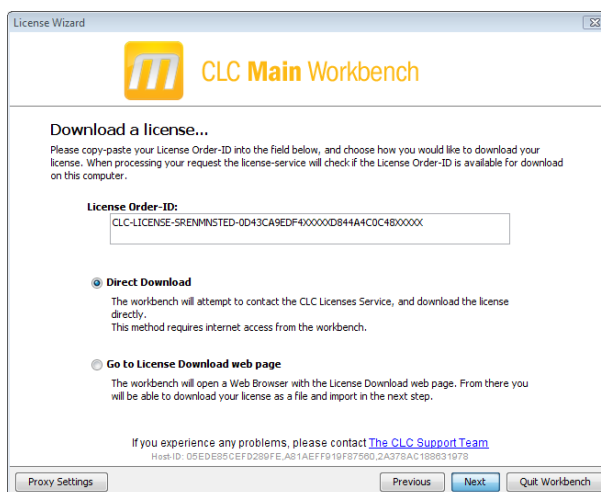
Figure 1.7: *Entering a license ID provided by CLC bio (the license ID in this example is artificial).*

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.

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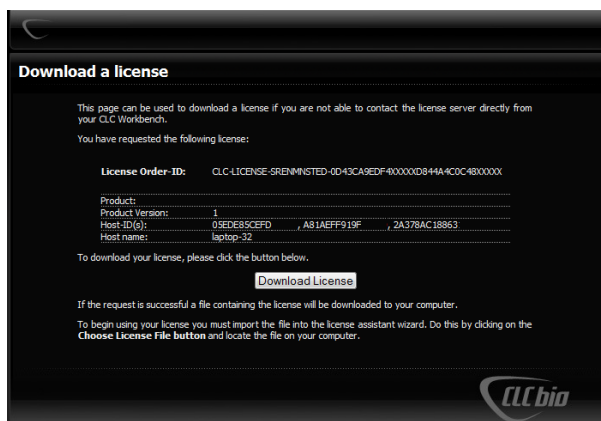
Figure 1.9: *The license web page where you can download a license.*

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 1.10.

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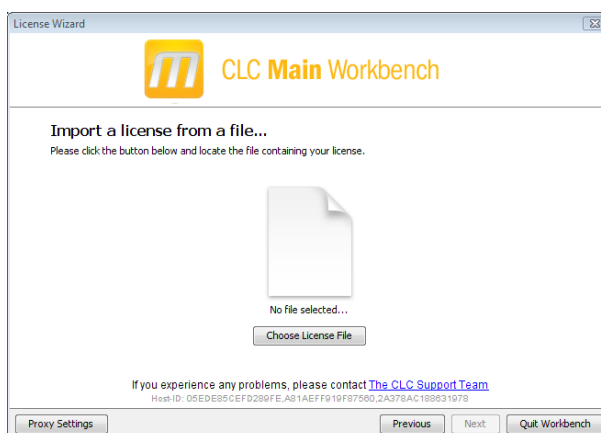
Figure 1.10: *Importing the license downloaded from the web site.*

Accepting the license agreement

Regardless of which option you chose above, you will now see the dialog shown in figure 1.11.
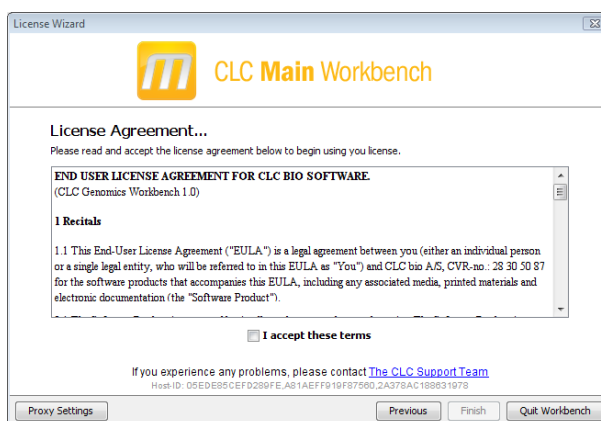
Figure 1.11: *Read the license agreement carefully.*

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

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**.

Accepting the license agreement

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

Please read the License agreement carefully before clicking **I accept these terms** and **Finish**.
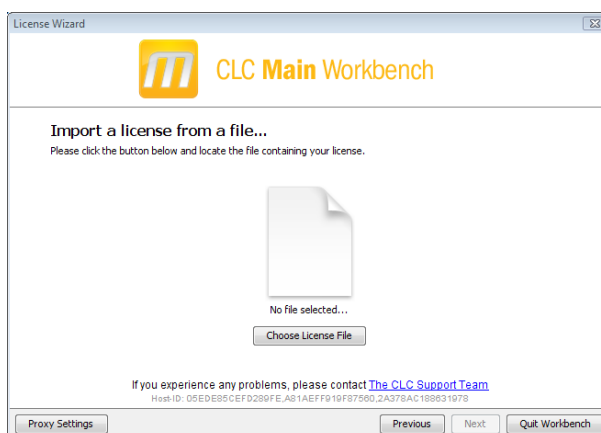
Figure 1.12: *Selecting a license file .*
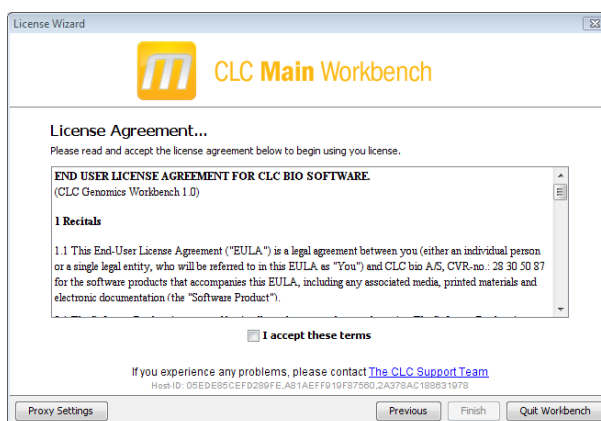
Figure 1.13: *Read the license agreement carefully.*

1.4.4 Upgrade license

If you already have used a previous version of *CLC Main Workbench*, and you are entitled to upgrading to the new *CLC Main Workbench 7.0*, select this option to get a license upgrade.

When you click **Next**, the workbench will search for a previous installation of *CLC Main 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:

- **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
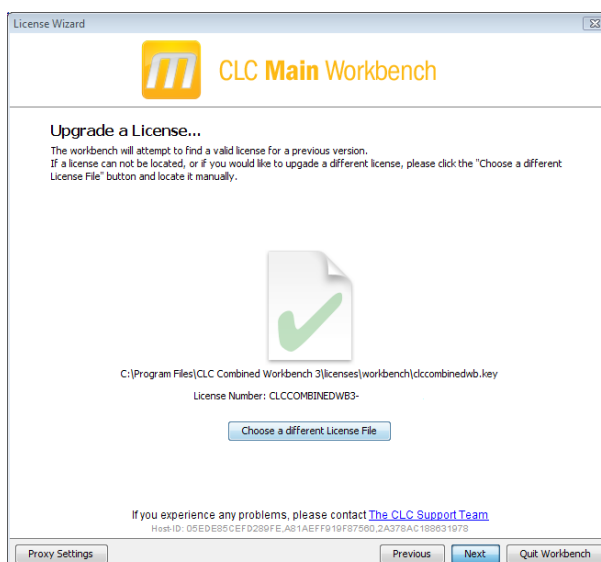
Figure 1.14: *An old license is detected.*

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.
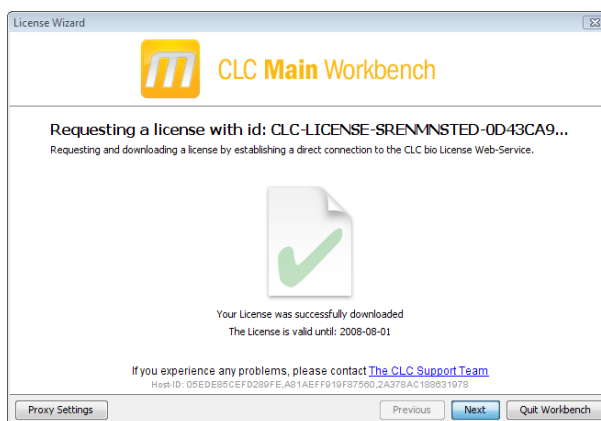
Figure 1.15: *A license has been downloaded.*

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

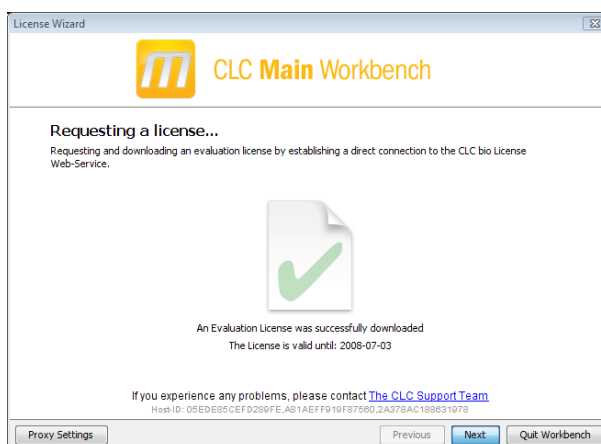Selecting the second option, *Go to license download web page*, opens the license web page as shown in 1.16.
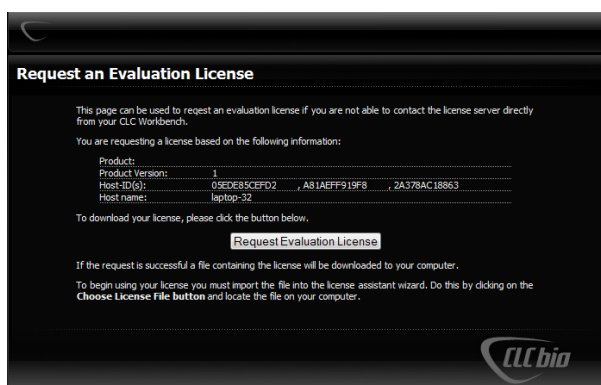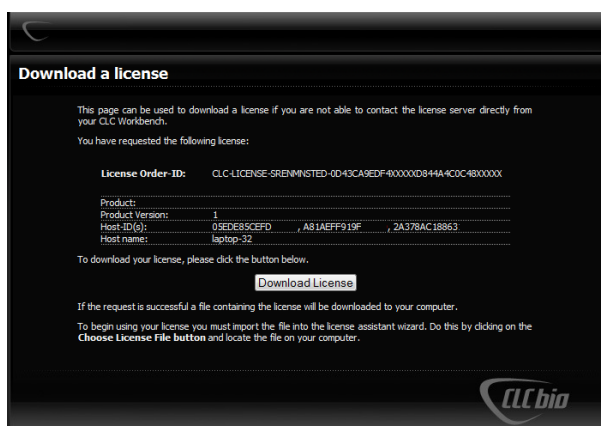
Figure 1.16: *The license web page where you can download a license.*

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

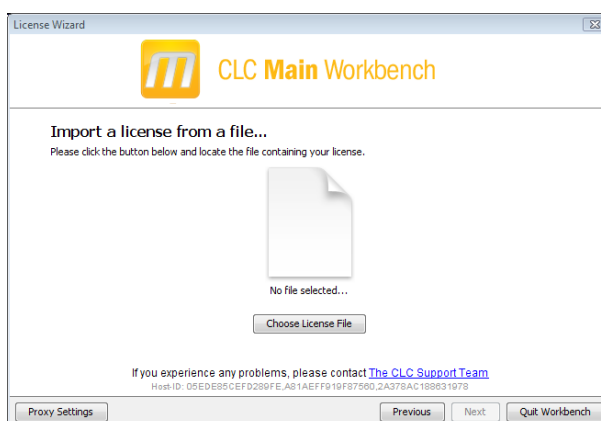Back in the Workbench window, you will now see the dialog shown in 1.17.
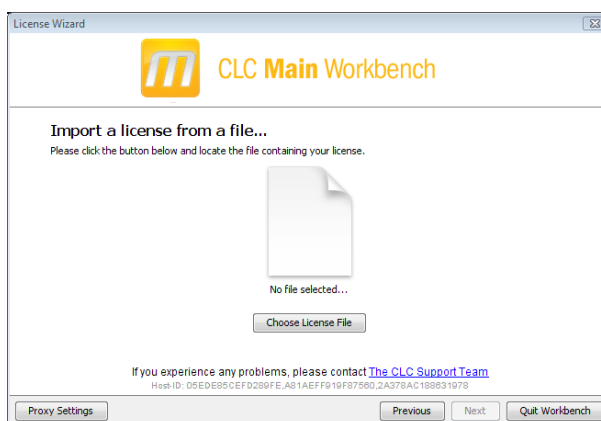
Figure 1.17: *Importing the license downloaded from the web site.*

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

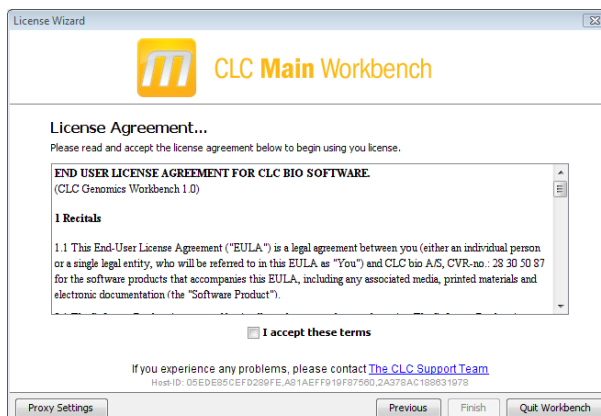Regardless of which option you chose above, you will now see the dialog shown in figure 1.18.

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

1.4.5 Configure license server connection

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

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.
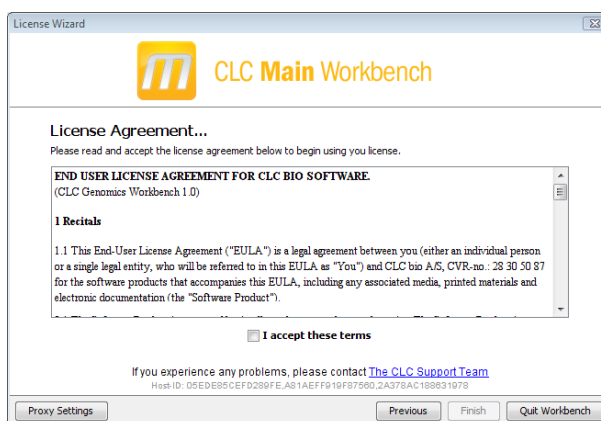
Figure 1.18: *Read the license agreement carefully.*
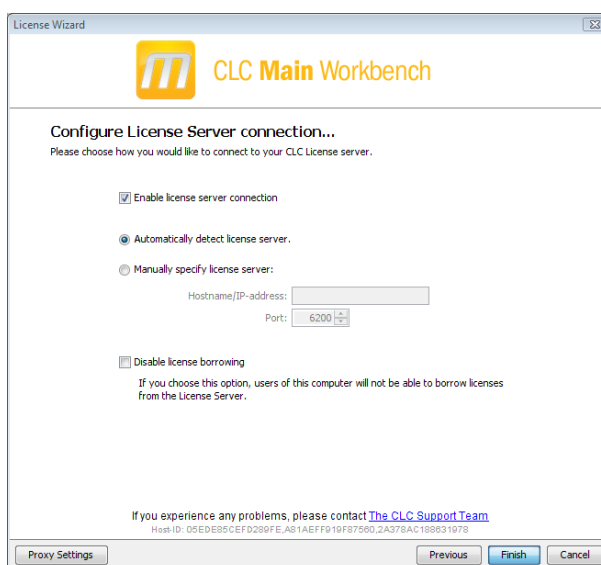
Figure 1.19: *Connecting to a license server.*

- **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 network license can only be used when you are connected to the license server. If you wish to use the *CLC Main Workbench* when you are not connected to the server, you can *borrow* a license. Borrowing a license means that you take one of the network licenses available on the

server and borrow it for a specified amount of time. During this time period, there will be one less network 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**

   and select the "Borrow License" tab to display the dialog in figure 1.20.
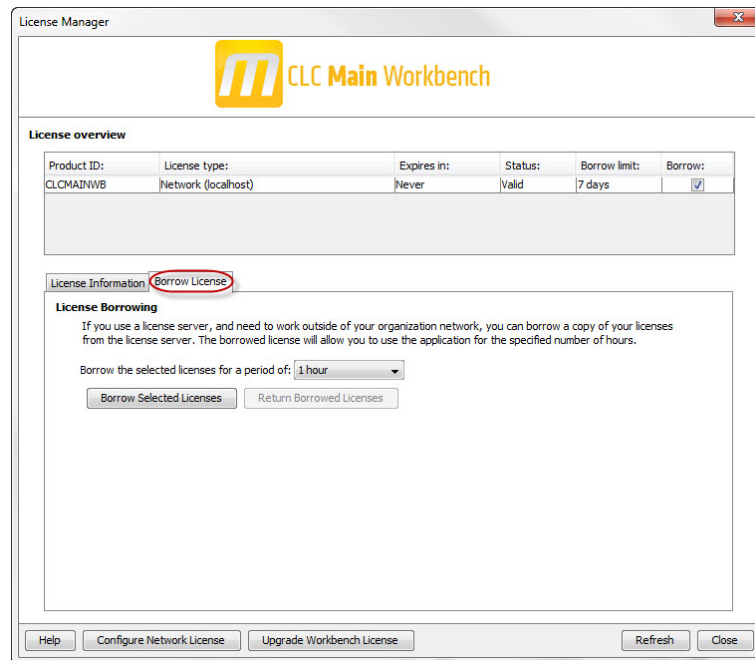
Figure 1.20: *Borrow a license.*

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 Main Workbench*.

5. When the borrow time period has elapsed, you have to connect to the license server again to use *CLC Main Workbench*.

6. When the borrow time period has elapsed, the license server will make the network 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 1.21 when you start the Workbench.
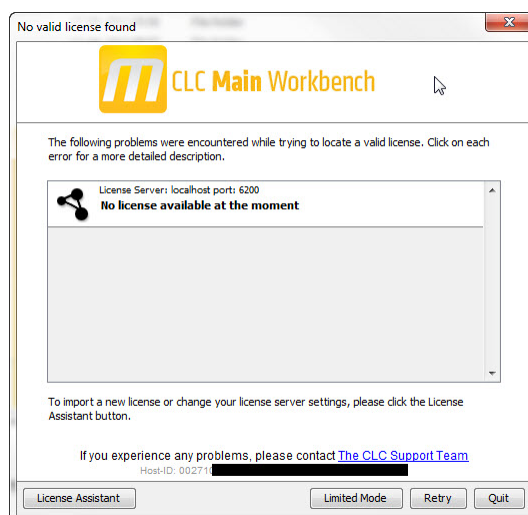
Figure 1.21: *No more licenses available on the server.*

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 which will start the Workbench with only a subset of features available but with the ability to access data.

If your connection to the license server is lost, you will see a dialog as shown in figure 1.22.
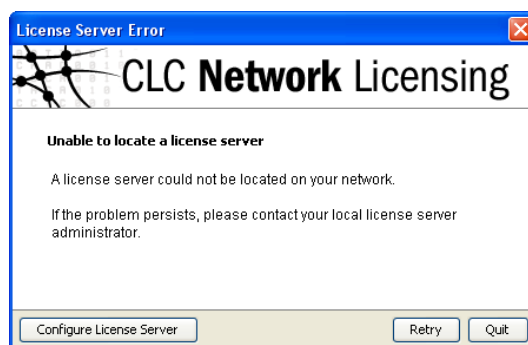
Figure 1.22: *Unable to contact license server.*

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 1.23.

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.
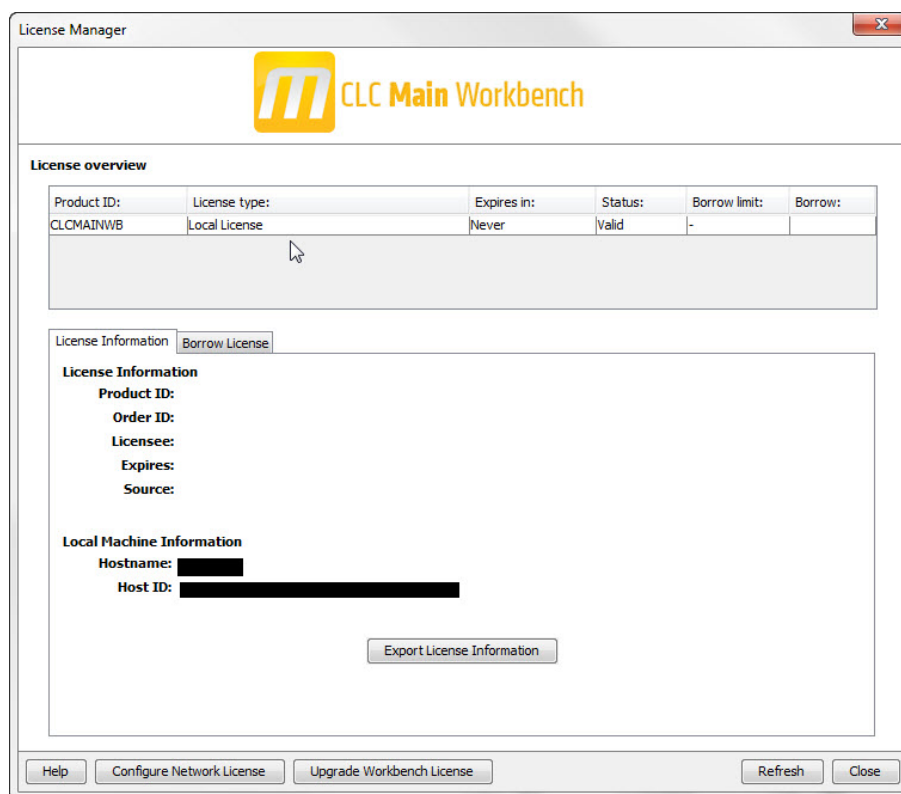
Figure 1.23: *The license manager.*

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

1.4.6 Download a license on a non-networked machine

To download a license for a machine that does not have direct access to the external network, you can follow the steps below:

- Install the *CLC Main Workbench* on the machine you wish to run the software on.

- Start up the software as an administrative user and find the host ID of the machine that you will run the CLC Workbench on. You can see the host ID the machine reported at the bottom of the License Manager window in grey text.

- Make a copy of this host ID such that you can use it on a machine that has internet access.

- Go to a computer with internet access, open a browser window and go to the relevant network license download web page:

- For Workbenches released from January 2013 and later, (e.g. the Genomics Workbench version 6.0 or higher, and the Main Workbench, version 6.8 or higher), please go to:

  https://secure.clcbio.com/LmxWSv3/GetLicenseFile

  For earlier Workbenches, including any DNA, Protein or RNA Workbench, please go to:

  http://licensing.clcbio.com/LmxWSv1/GetLicenseFile

It is vital that you choose the license download page appropriate to the version of the software you plan to run.

- Paste in your license order ID and the host ID that you noted down in the relevant boxes on the webpage.

- Click 'download license' and save the resulting .lic file.

- Open the Workbench on your non-networked machine. In the Workbench license manager choose 'Import a license from a file'. In the resulting dialog click 'choose license file' to browse the location of the .lic file you have just downloaded.

  If the License Manager does not start up by default, you can start it up by going to the Help menu and choosing License Manager.

- Click on the **Next** button and go through the remaining steps of the license manager wizard.

1.4.7 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 all new high-throughput sequencing platforms such as Roche-454, Illumina and SOLiD in addition to Sanger reads and hybrid data.

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 plugins. The plugins are fully integrated with the CLC Workbenches and the Viewer and provide an easy way to customize and extend their functionalities.

In April 2012, *CLC Protein Workbench*, *CLC DNA Workbench*and *CLC RNA Workbench* were discontinued. All customers with a valid license for any of these products were offered an upgrade to the *CLC Main Workbench*.

In February 2014, CLC bio expanded the product repertoire with the release of *CLC Drug Discovery Workbench*, a product that enables studies of protein-ligand interactions for drug discovery.

1.5.1 New program feature request

The CLC team is continuously improving the *CLC Main Workbench* with our users' interests in mind. We welcome all requests and feedback from users, as well as suggestions for new features or more general improvements to the program. To contact us via the Workbench, please go to the menu option:

> **Help | Contact Support**

1.5.2 Getting help

If you encounter a problem or need help understanding how the *CLC Main Workbench* works, and the license you are using is covered by our Maintenance,UpgradesandSupport(MUS) program, you can contact our customer support via the Workbench by going to the menu option:

> **Help | Contact Support**

This will open a dialog to enter your contact information and a text field for entering the question or problem you have.

You can also attach small datasets, if this helps explain the problem or you believe it will help in troubleshooting the problem.

When you send a support request this way, it will include technical information about your installation that usually helps when troubleshooting. It also includes your license information so that you do not have to look this up yourself. Our support staff will reply to you by email.

Further information about Maintenance, Upgrades and Support (MUS) program can be found online at https://www.clcbio.com/support/maintenance-support-program/.

Information about MUS cover on particular licenses can be found by https://secure.clcbio.com/myclc/login.

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 Main Workbench* again (without pressing Shift.

1.5.3 CLC Sequence Viewer vs. Workbenches

The advanced analyses of the commercial workbenches, *CLC Genomics Workbench* and *CLC Main Workbench* are not present in *CLC Sequence Viewer*. Likewise, some advanced analyses are available in *CLC Genomics Workbench* but not in *CLC Main Workbench*. All types of basic and advanced analyses are available in *CLC Genomics 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 Main 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.

Tutorials describing hands-on examples of how to use the individual tools and features of the *CLC Main Workbench* can be found at http://www.clcbio.com/support/tutorials/. 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 video tutorials and other online presentations here: http://www.clcbio.tv/.

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 1.24.
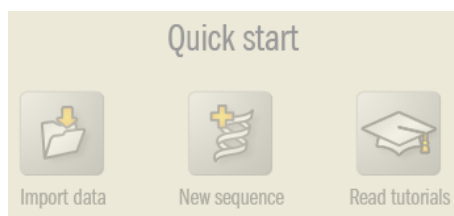
Figure 1.24: *Quick start short cuts, available in the background of the workspace.*

The function of the 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 Main Workbench* includes an example data set.

When downloading *CLC Main 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 **Import Example Data** ( ) in the **Help** menu of the program. This imports 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 13 for more about importing data.

1.7 Plugins

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

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

1.7.1 Installing plugins

Plugins are installed using the plugin manager[3]:

> **Help in the Menu Bar | Plugins and Resources... ( )**

> or **Plugins ( ) in the Toolbar**

The plugin manager has four tabs at the top:

- **Manage Plugins.** This is an overview of plugins that are installed.

- **Download Plugins.** This is an overview of available plugins 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.

---

[3]In order to install plugins 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.

To install a plugin, click the **Download Plugins** tab. This will display an overview of the plugins that are available for download and installation (see figure 1.25).
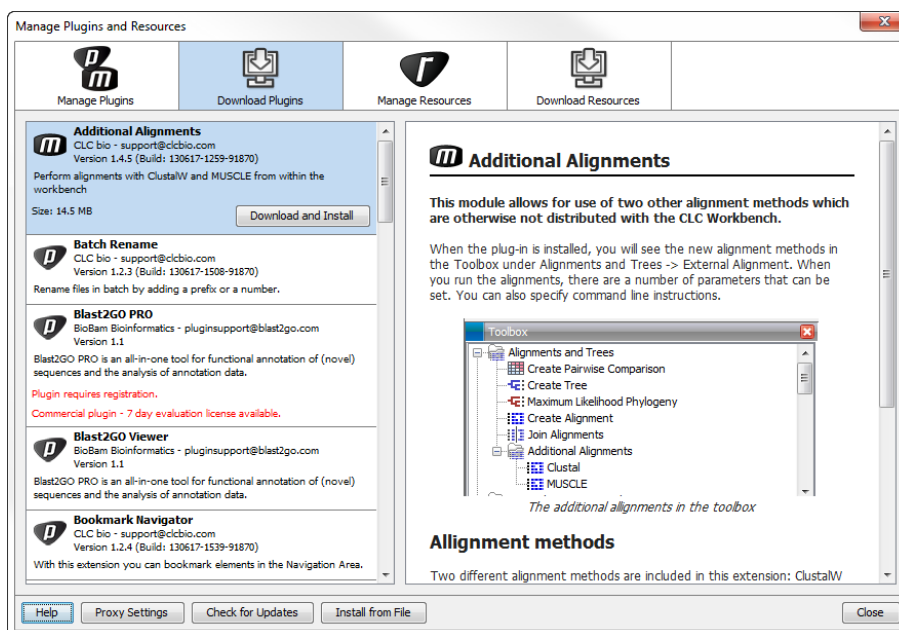
Figure 1.25: *The plugins that are available for download.*

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

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

If the plugin 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 plugin. The plugin 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 Main Workbench*. The plugin will not be ready for use until you have restarted.

1.7.2 Uninstalling plugins

Plugins are uninstalled using the plugin manager:

> **Help in the Menu Bar | Plugins and Resources... ( )**

> or **Plugins ( ) in the Toolbar**

This will open the dialog shown in figure 1.26.

The installed plugins are shown in this dialog. To uninstall:

> **Click the plugin | Uninstall**

If you do not wish to completely uninstall the plugin but you don't want it to be used next time you start the Workbench, click the **Disable** button.
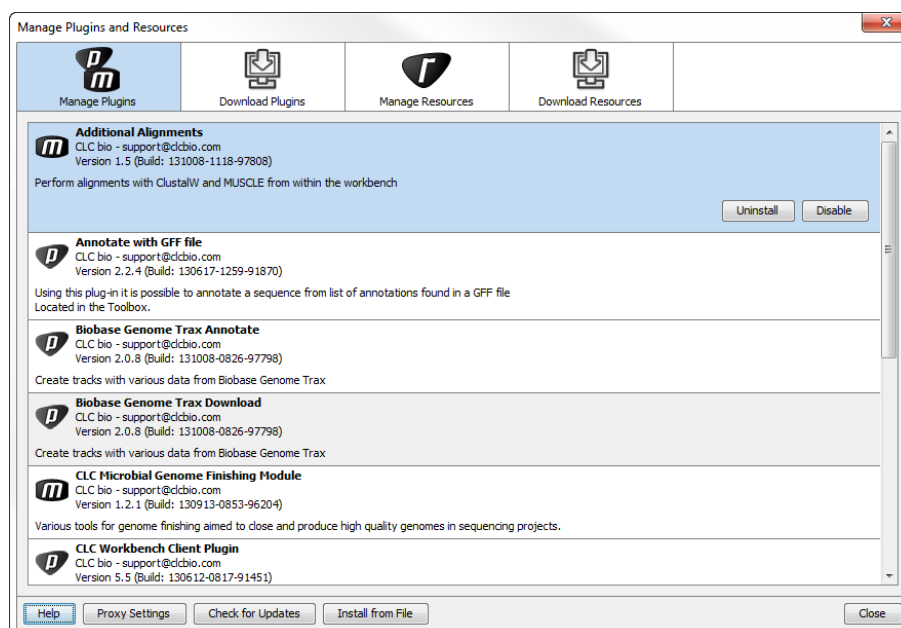
Figure 1.26: *The plugin manager with plugins installed.*

When you close the dialog, you will be asked whether you wish to restart the workbench. The plugin will not be uninstalled until the workbench is restarted.

1.7.3   Updating plugins

If a new version of a plugin is available, you will get a notification during start-up as shown in figure 1.27.

In this list, select which plugins you wish to update, and click **Install Updates**. If you press **Cancel** you will be able to install the plugins later by clicking **Check for Updates** in the Plugin manager (see figure 1.26).

1.7.4   Resources

Resources are downloaded, installed, un-installed and updated the same way as plugins. Click the **Download Resources** tab at the top of the plugin manager, and you will see a list of available resources (see figure 1.28).

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

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

1.8   Network configuration

If you use a proxy server to access the Internet you must configure *CLC Main Workbench* to use this. Otherwise you will not be able to perform any online activities (e.g. searching GenBank).

CLC Main Workbench supports the use of a HTTP-proxy and an anonymous SOCKS-proxy.
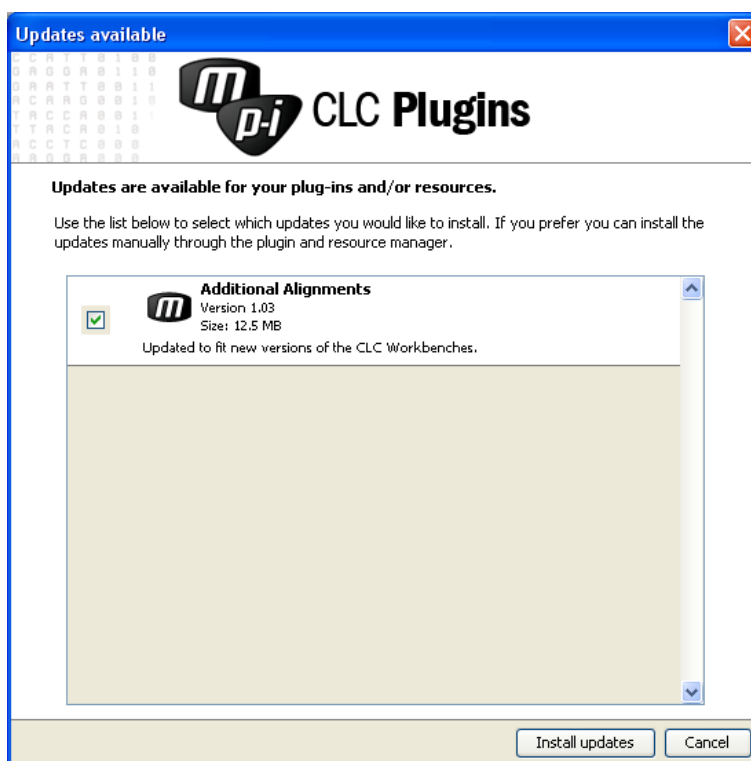
Figure 1.27: *Plugin updates.*
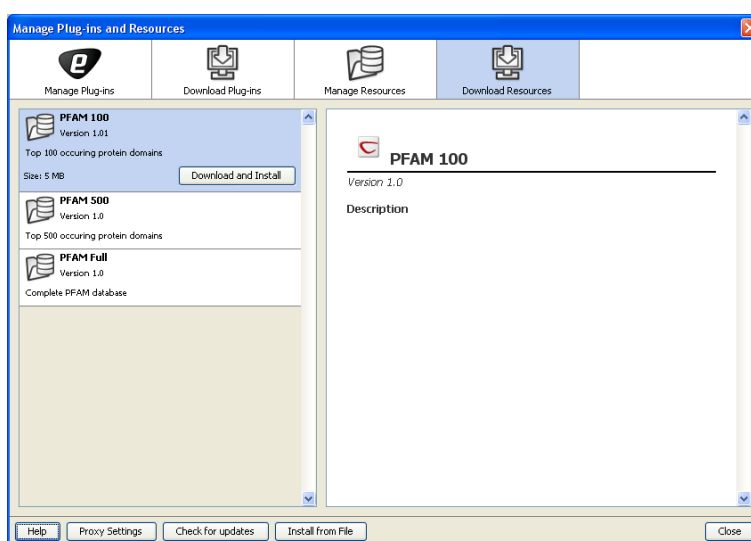
Figure 1.28: *Resources available for download.*

To configure your proxy settings, open *CLC Main Workbench*, and go to the **Advanced**-tab of the **Preferences** dialog (figure 1.29) 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 Main Workbench* only supports the use of a SOCKS-proxy that does not require authorization.

You can select whether the proxy should be used also for FTP and HTTPS connections.

Exclude hosts can be used if there are some hosts that should be contacted directly and not
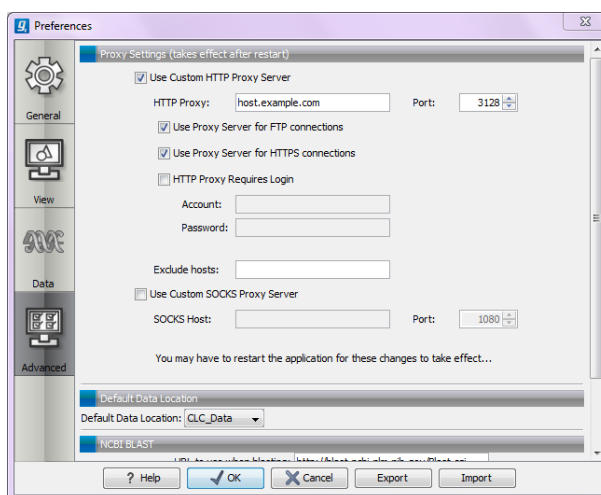
Figure 1.29: *Adjusting proxy preferences.*

through the proxy server. The value can be a list of hosts, each separated by a `|`, and in addition a wildcard character `*` can be used for matching. For example: `*.foo.com|localhost`.

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 to the *CLC Main 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 molecular modeling and 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 Main Workbench* and provide more general knowledge of molecular modeling and 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**)

1.10 Latest improvements

CLC Main Workbench is under constant development and improvement. A detailed list that includes a description of new features, improvements, bugfixes, and changes for the current version of *CLC Main Workbench* can be found at:

http://www.clcbio.com/products/latest-improvements-main-workbench/

Chapter 2

Tutorials

Contents

This chapter contains tutorials representing some of the features of *CLC Main 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 Main Workbench*. **Watch video tutorials at `http://www.clcbio.tv`.**

2.1   Tutorial: Getting Started

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

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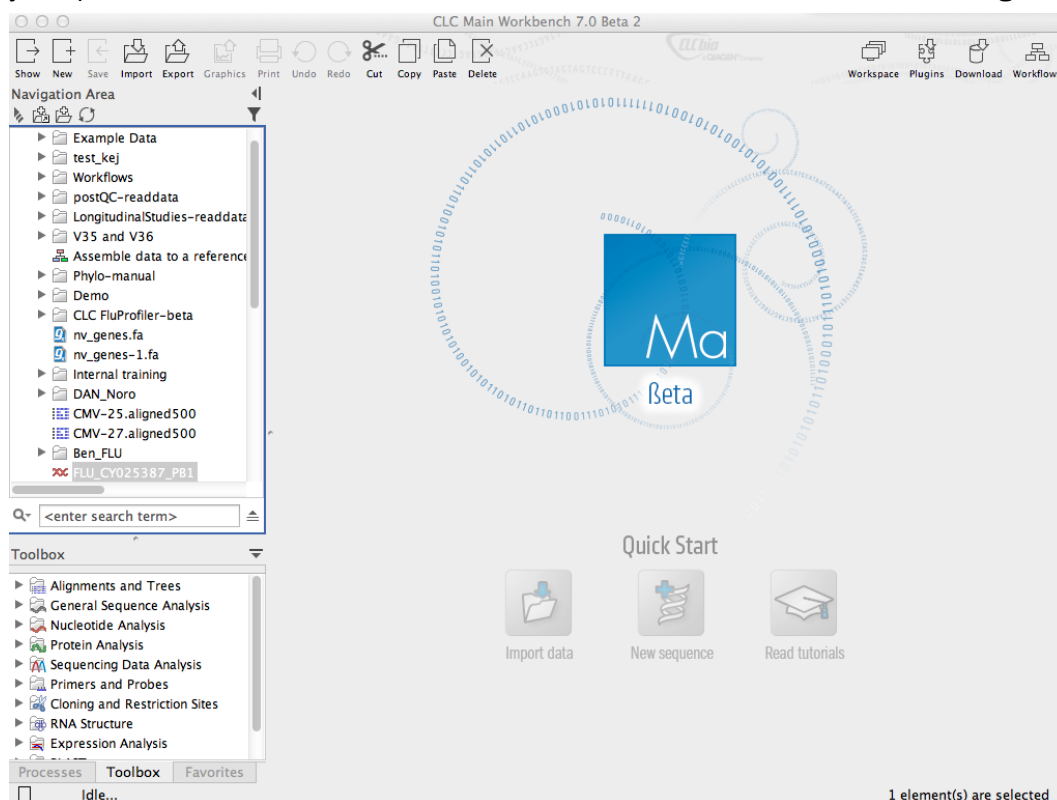
Figure 2.1: *The user interface as it looks when you start the program for the first time. (Mac version of* **CLC Main Workbench***. The interface is similar for Windows and Linux.)*

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 Main 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. (see section 2.1.2).

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 Main 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 Main Workbench* is stored.

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

> **File** | **New** | **Folder** (⌺)
> or   **Ctrl + Shift + N (⌘ + Shift + N on Mac)**

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

2.1.2   Import data

As an example, first generation sequence data as well as high throughput sequencing data can be downloaded from http://www.clcbio.com/downloads under "EXAMPLE DATA", Roche/454 pyrosequencing genome data from E. coli commensal strain K-12. The NC_010473.gbk (GenBank format) can be imported by all types of CLC Workbenches, while import of the high throughput sequencing data requires specialised import actions. (This "EXAMPLE DATA" 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.)

The sequence data 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 NC_010473.gbk (GenBank format) result looks like figure 2.2 while the high throughput data looks like figure 2.3.

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.

2.2   Tutorial: View a DNA 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 2.4).

As default, *CLC Main Workbench* displays a sequence with annotations (colored arrows on the sequence like the green promoter region annotation in figure 2.4) and zoomed to see the residues.

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

---

[1]If you have downloaded the example data, this will be placed as a folder in *CLC_Data*
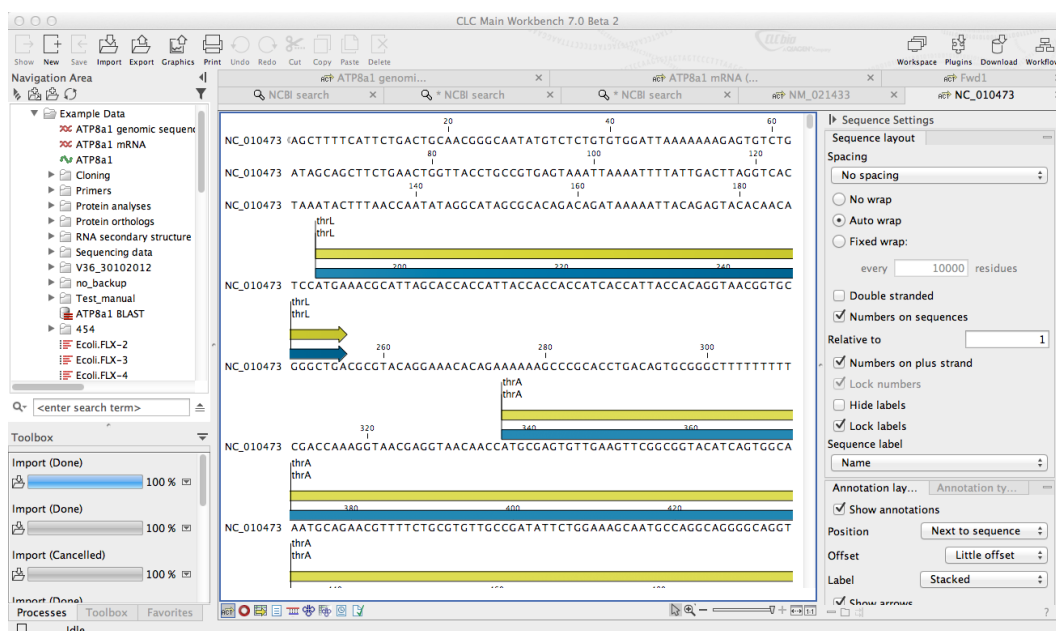
Figure 2.2: *The NC_010473.gbk (GenBank format) file is imported and opened.*
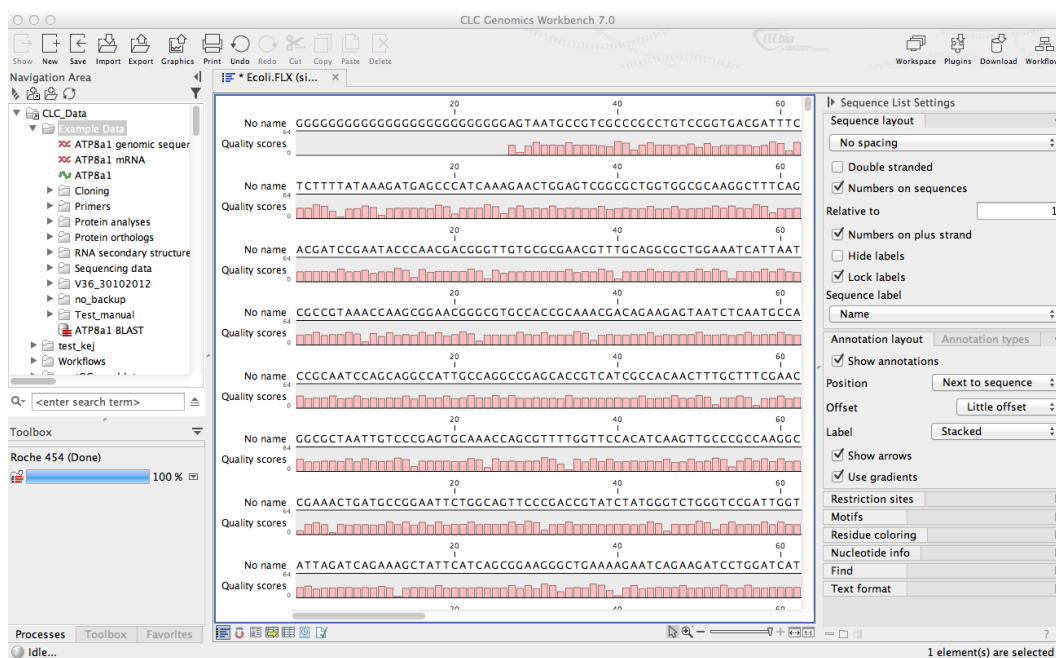
Figure 2.3: *The NC_010473 high throughput data imported and opened.*

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:
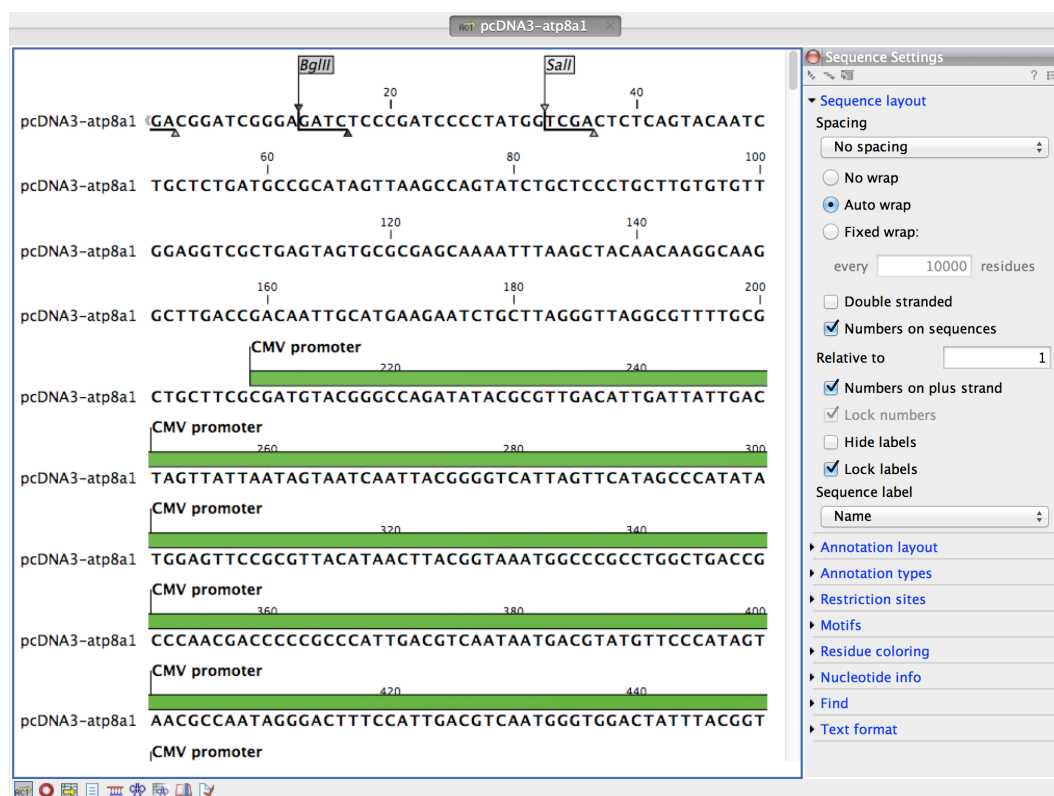
Figure 2.4: *Sequence pcDNA3-atp8a1 opened in a view.*

> **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 2.5.

You may want to change the text size in the top panel to see more of the sequence. Scroll down in the **Sequence Settings** panel to **Text format** and change text size to **Tiny** or **Small**. You can also resize the panels relative to each other by clicking and dragging the separator line between them.

Make a selection on the circular sequence (remember to switch to the **Selection** (🔖) tool in the tool bar). Note that this selection is also reflected in the linear view above, and that the selection coordinates appear at the bottom right corner of the screen (in figure 2.5 the **Ampicillin ORF** was selected).

You can open a third view of just the selected part of the sequence by right-clicking anywhere in the highlighted sequence text in the top panel and choosing **Open Selection in New View** as shown in figure 2.6.

Click and drag the new tab from the bottom panel to the top one, next to the existing linear view.

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
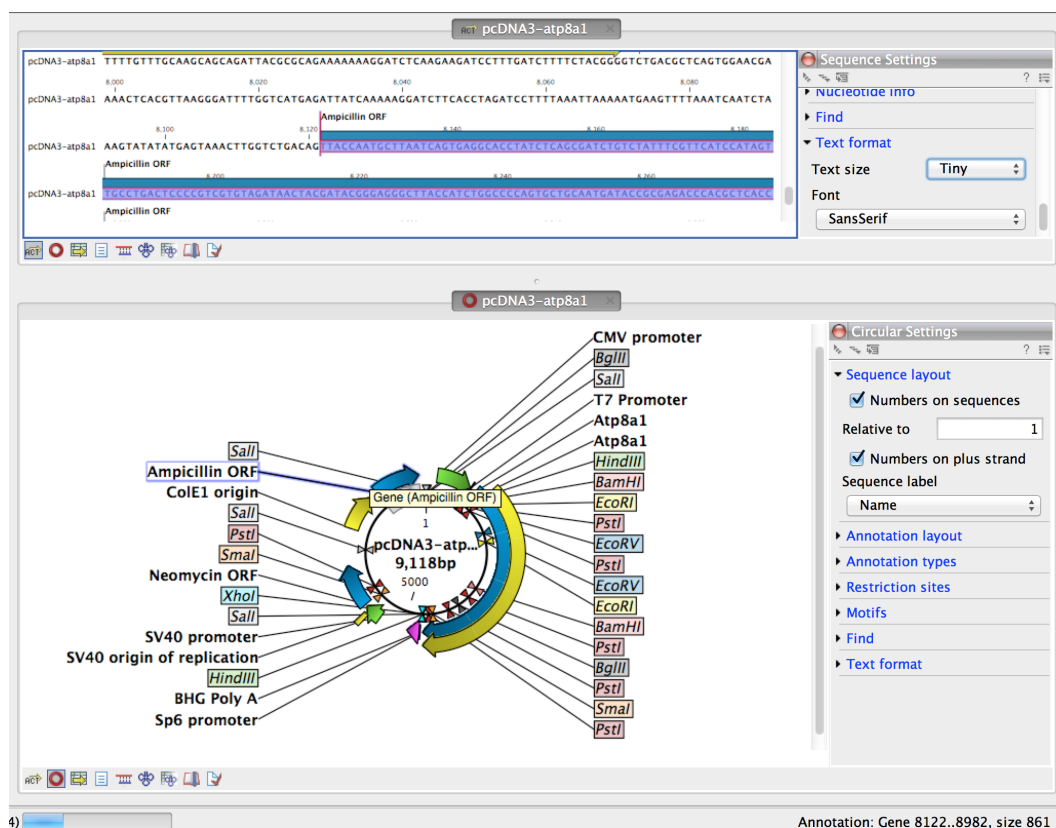
Figure 2.5: *The resulting two views which are split horizontally.*

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 alignment is automatically wrapped to fit the width of the view (shown in figure 2.7).

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".

Click on the **Annotation Types** tab. Here you will see a list of the types annotation that are carried by the sequences in the alignment (see figure 2.8).

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.
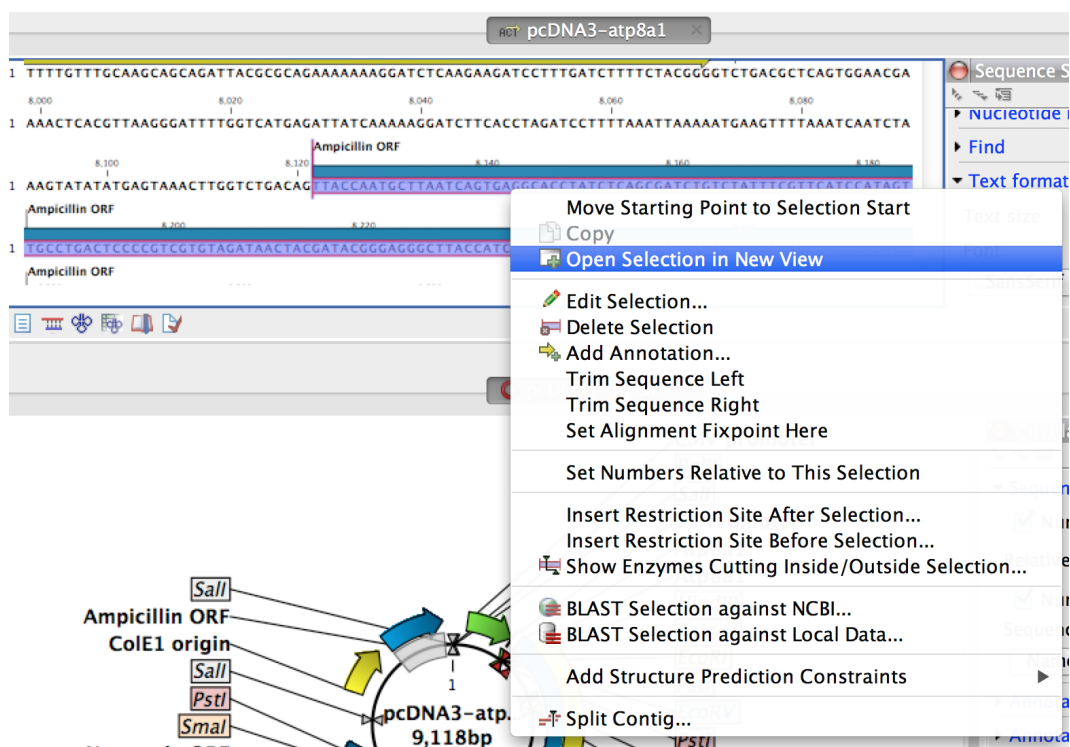
Figure 2.6: *Creating a new view panel for just the selected sequence.*
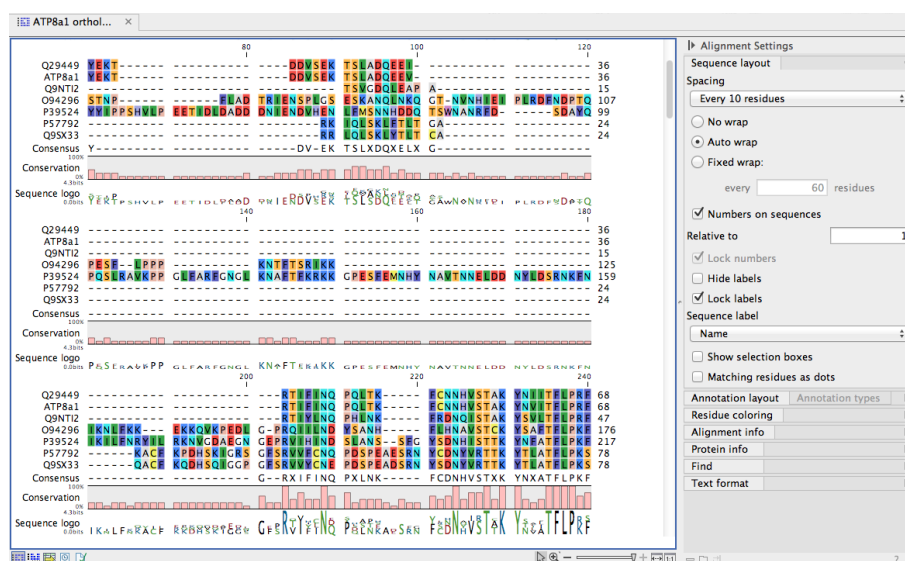
Figure 2.7: *The protein alignment as it looks when you open it with background color according to the Rasmol color scheme and automatically wrapped.*

2.3.1 Saving the settings in the Side Panel

Now the alignment should look similar to figure 2.9.

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 bottom of the **Side Panel** and click **Save Alignment View Settings** (see figure 2.10).
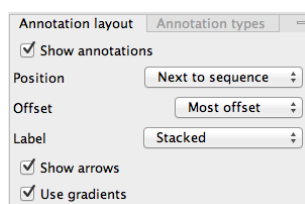
Figure 2.8: *The Annotation Layout and the Annotation Types tabs in the Side Panel.*
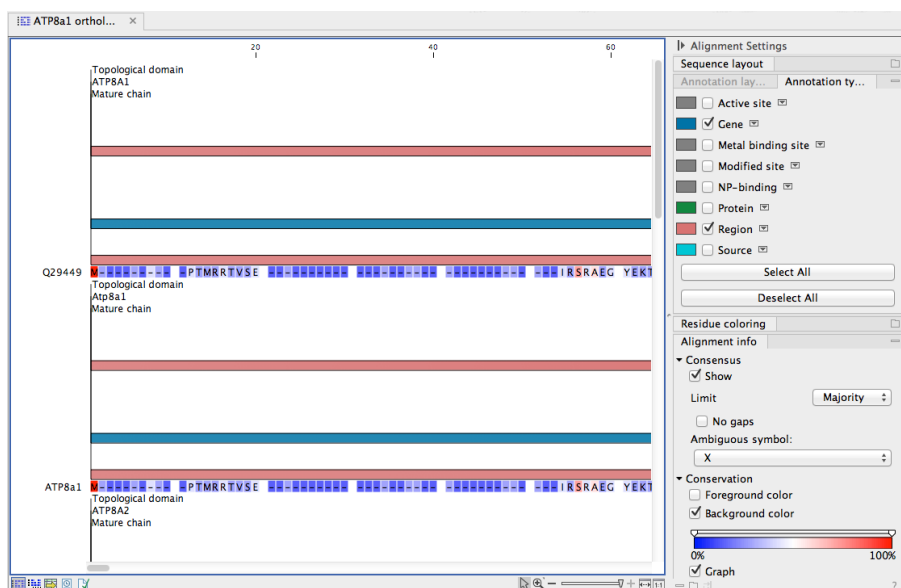
Figure 2.9: *The alignment when all the above settings have been changed.*
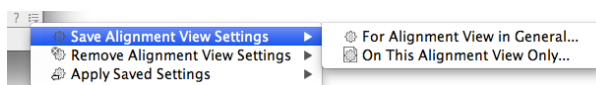
Figure 2.10: *Saving the settings of the Side Panel either generally or this particular alignment only.*

This will open the dialog shown in figure 2.11.
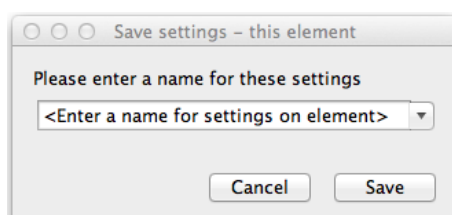
Figure 2.11: *Dialog for saving the settings of the Side Panel.*

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 Remove alignment view settings

When you click the **Save/Restore Settings** button (⊟) and select **Remove Alignments View Settings**, you can choose whether this should be applied generally or on this alignment view only

(see figure 2.12).
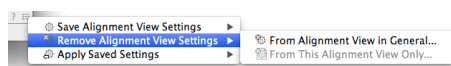
Figure 2.12: *Menu for removing saved settings.*

This will open the dialog shown in figure 2.13 and allow you to remove specific settings.
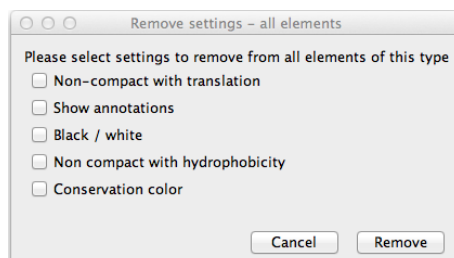
Figure 2.13: *Menu for removing saved settings.*

2.3.3 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 Main Workbench* has created for you (see figure 2.14).
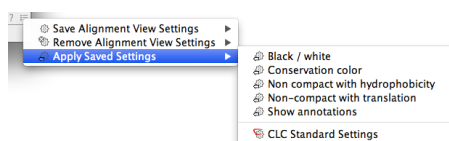
Figure 2.14: *Menu for applying saved settings.*

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.

2.4   Tutorial: GenBank Search and Download

The 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 the 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:

> **Download | 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

Figure 2.15: *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: 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).

2.5.1 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 2.16.
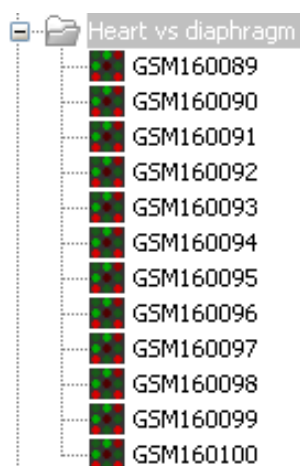
Figure 2.16: *12 microarrays have been imported.*

2.5.2 Grouping the samples

The next step is to tell the *CLC Main 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 Main 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 | Transcriptomics Analysis ( )| Set Up Experiment ( )

Select the 12 arrays that you have imported (see figure 2.17).
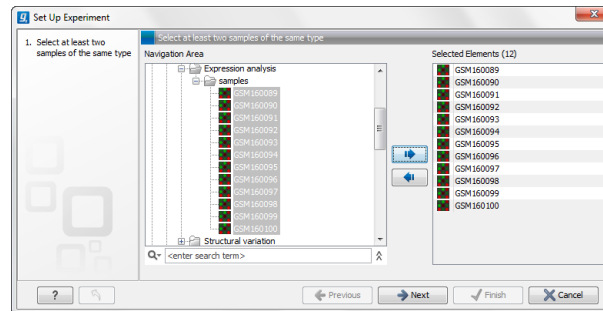
Figure 2.17: *Select the 12 microarrays that have been imported.*

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 2.18.
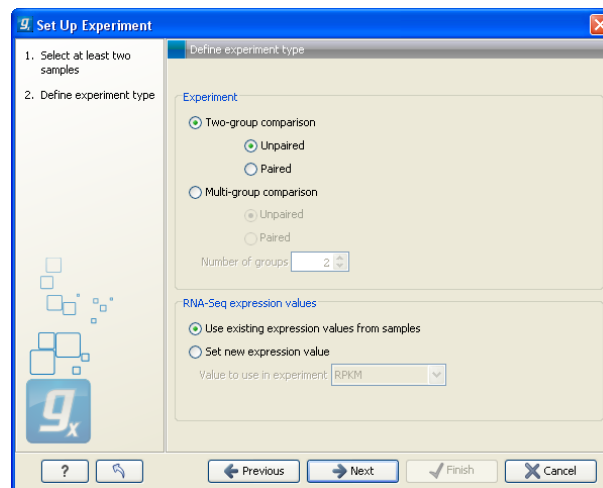
Figure 2.18: *Defining the number of groups.*

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 2.19.
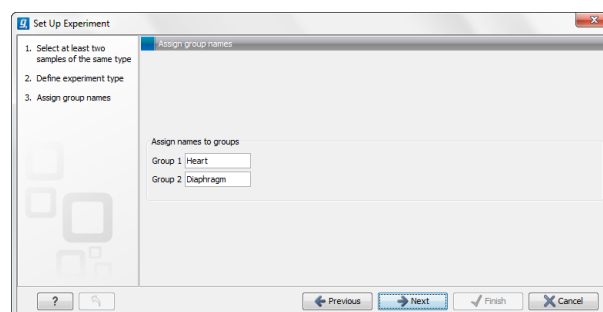
Figure 2.19: *Naming the groups.*

Name the first group **Heart** and the second group **Diaphragm** and click **Next** (see figure 2.20).
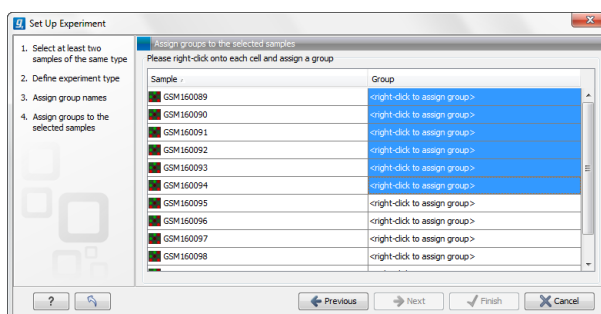
Figure 2.20: *Assigning the samples to groups.*

Here you see a list of all the samples you chose in figure 2.17. 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 in the **Navigation Area** is copied into the experiment, so they now exist independently of each other.

2.5.3   The experiment table

Once it is created, the experiment will be opened in a table as shown in figure 2.21.

Figure 2.21: *The experiment table.*

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.

2.6 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.

2.6.1 Transformation

First we inspect to what extent the variance in expression values depends on the mean. For this we create an **MA Plot**:

Toolbox | Transcriptomics Analysis ( ) | General Plots | Create MA Plot ( )

Since the MA plot compares two samples, we will start out selecting two of the 12 arrays and click **Finish**. This will show a plot similar to the one shown in figure 2.22.
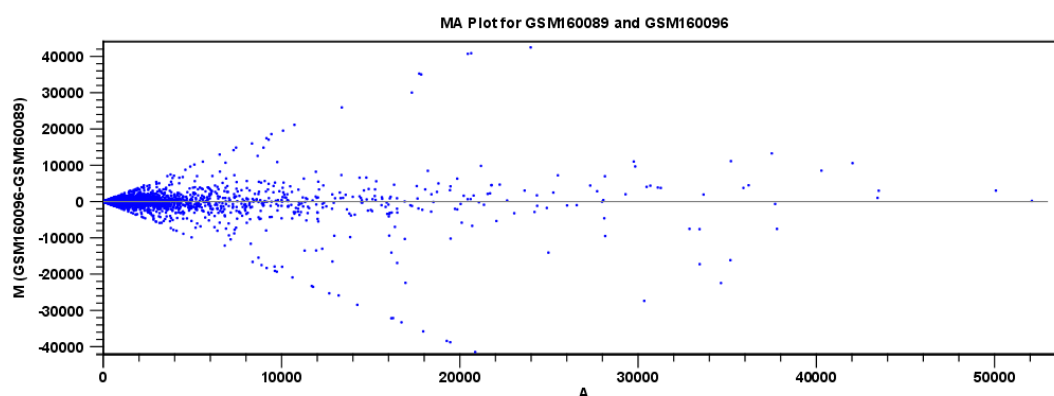
Figure 2.22: *MA plot before transformation.*

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 2.22 it is clear that the variance increases with the mean. To remove some of the dependency, we want to **transform** the data:

Toolbox | Transcriptomics Analysis ( ) | Transformation and Normalization | Transform ( )

Select the same two arrays as 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 2.23).
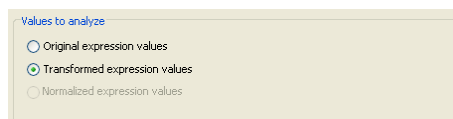
Figure 2.23: *Select the transformed expression values.*

Select the transformed values. You will see that these three selection boxes; Original, Trans-

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 2.24.
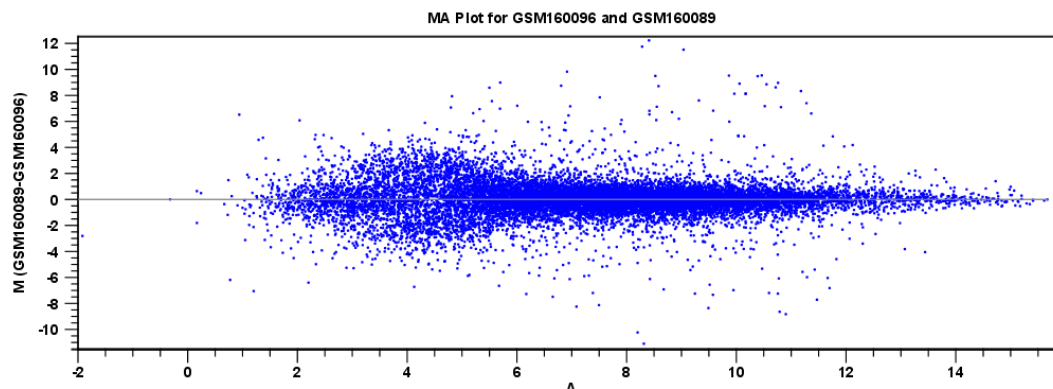
Figure 2.24: *MA plot after transformation.*

The much more symmetric and even spread indicates that the dependence 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. Hence, we need to transform the expression values within the entire experiment, as we will use the transformed data in the further analysis. To transform the entire experiment:

> **Toolbox** | **Transcriptomics Analysis** (🖼) | **Transformation and Normalization** | **Transform** (📈)

Select the experiment created in the first part of this tutorial series, click **Next**, choose **Log 2** transformation and click **Finish**.

If you open the table, you will see that all the samples have an extra column with transformed expression values (figure 2.25).

There is also an extra column for transformed group means and transformed IQR.

2.6.2 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** | **Transcriptomics 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 2.26.

This plot looks very good because none of the samples stands out from the rest. If you compare

Figure 2.25: *Transformed expression values have been added to the table.*
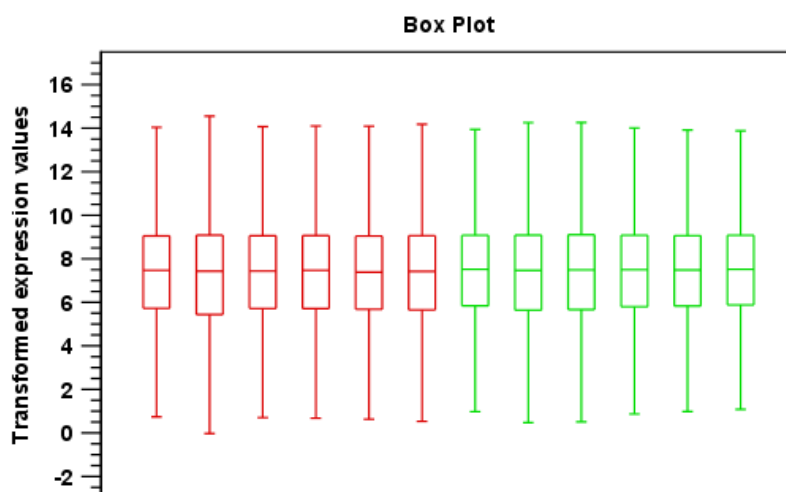
Figure 2.26: *A box plot of the 12 samples in the experiment, colored by group.*

this plot to the one shown in figure 2.27 from another data set, you can see the difference.

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.

2.6.3 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** | **Transcriptomics Analysis** ( ) | **Quality Control** | **Principal Component Analysis ( )**
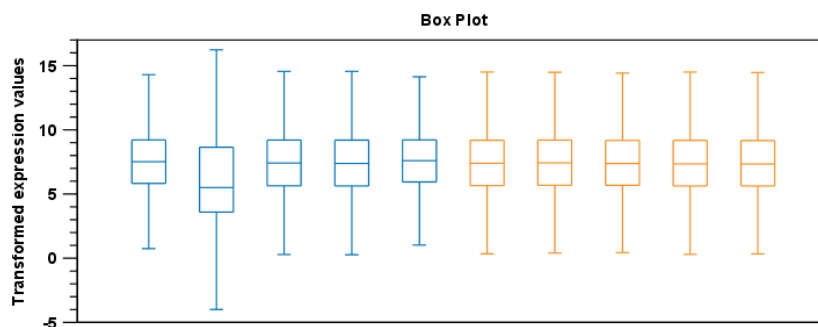
Figure 2.27: *A box plot showing one sample that stands out from the rest.*

Select the experiment and click **Next**. **Finish**. This will create a PCA plot as shown in figure 2.28).
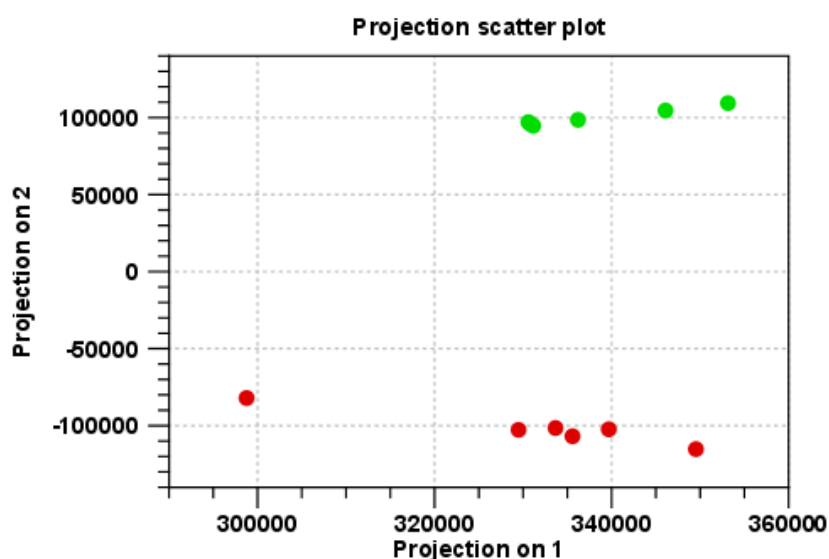
Figure 2.28: *A principal component analysis colored by group.*

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 2.29).
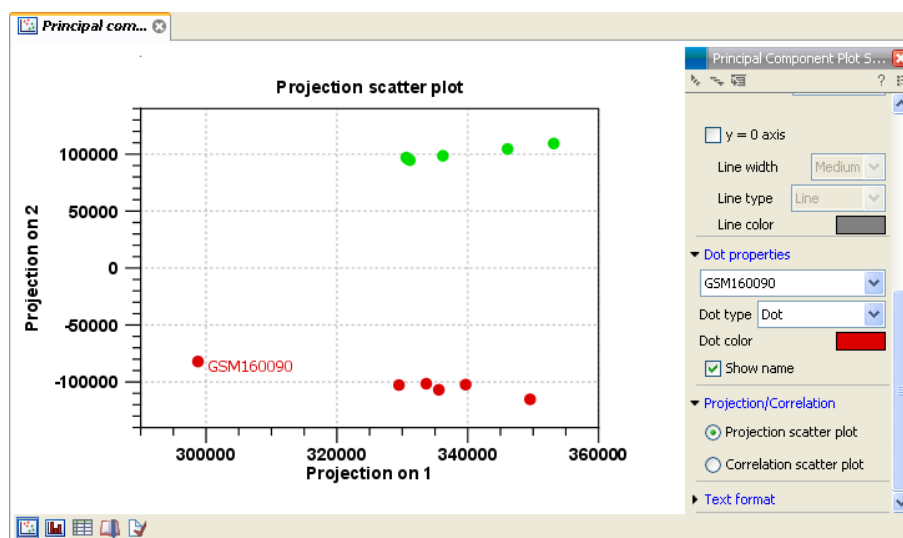
Figure 2.29: *Naming the outlier.*

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** | **Transcriptomics 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 2.30).
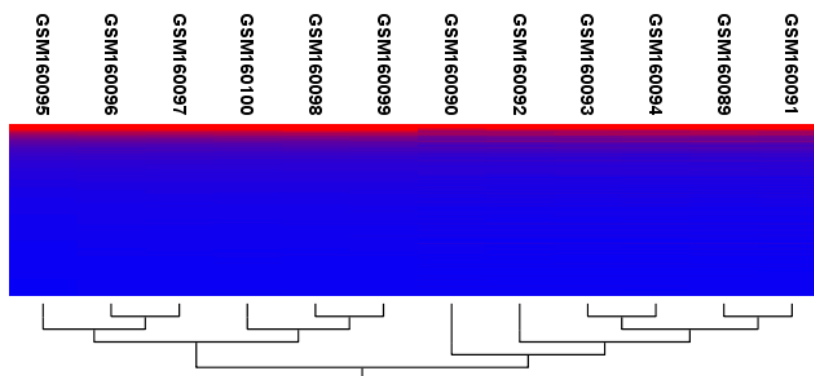
Figure 2.30: *Sample clustering.*

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. You can use the buttons at the bottom of the editor to

switch between different views in figure 2.31.

Figure 2.31: *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.

2.7 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.

2.7.1 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 | Transcriptomics 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 2.32.
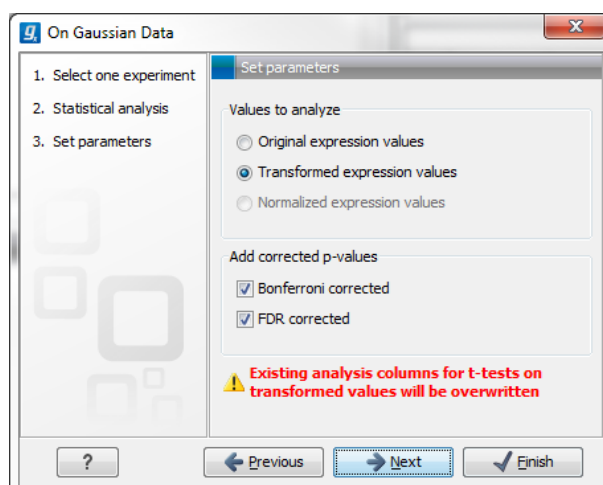
Figure 2.32: *Statistical analysis.*

As shown in figure 2.32 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 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 2.33).

Figure 2.33: *FDR p-values compared to Bonferroni-corrected p-values.*

2.7.2 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 **t-test: Heart vs Diaphragm transformed values - 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 2.34).

Figure 2.34: *Filtering on FDR p-values.*

You can see that 1471 genes fulfilled this criterion (marked with a red circle).

2.7.3   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 2.35.
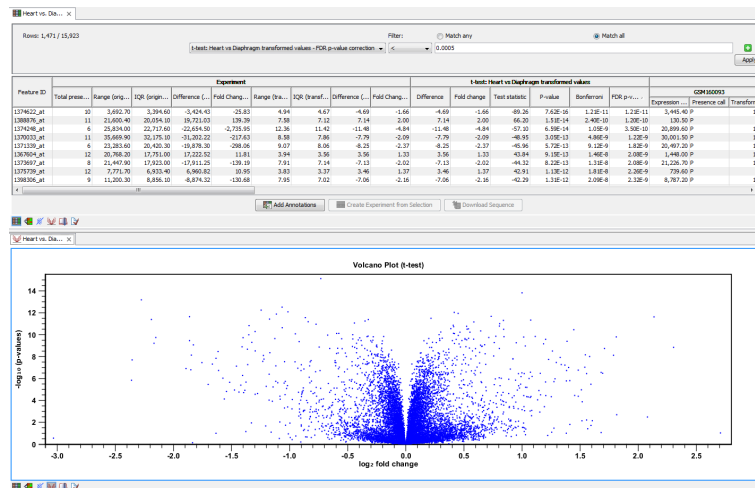
Figure 2.35: *Split view of volcano plot and experiment table.*

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 2.36).
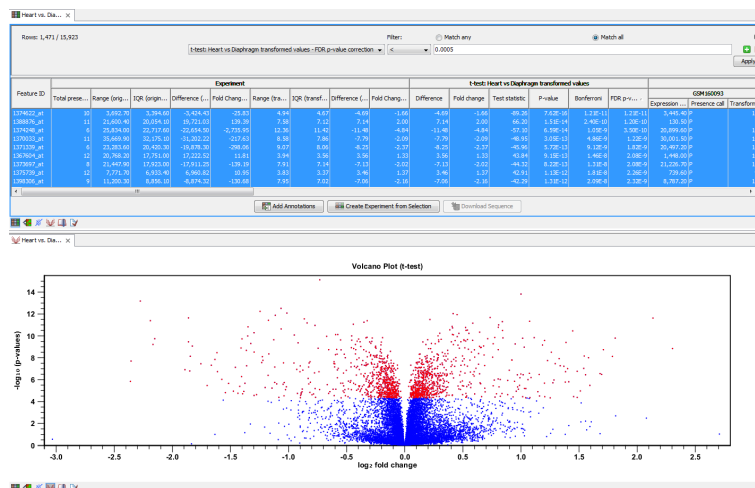
Figure 2.36: *Volcano plot where selected dots are colored red.*

2.7.4 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 2.37.
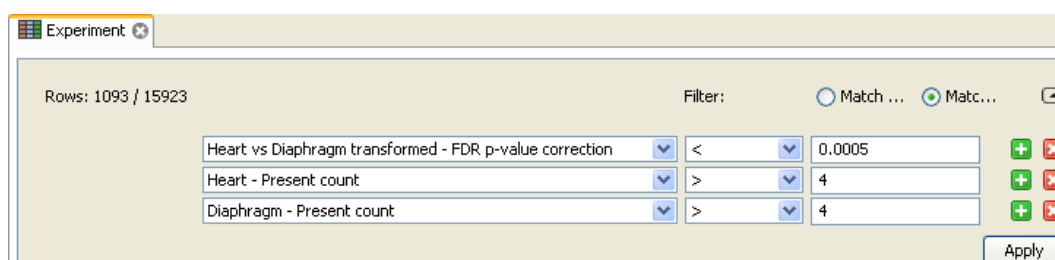
Figure 2.37: *Filtering genes where at least 5 out of 6 calls in each group are present.*

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 2.38.
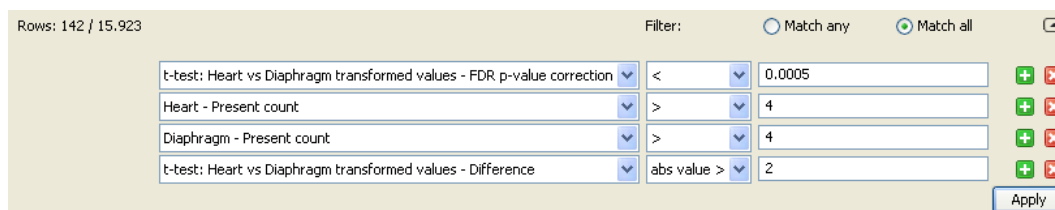
Figure 2.38: *The absolute value of group mean difference should be larger than 2.*

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 2.39.
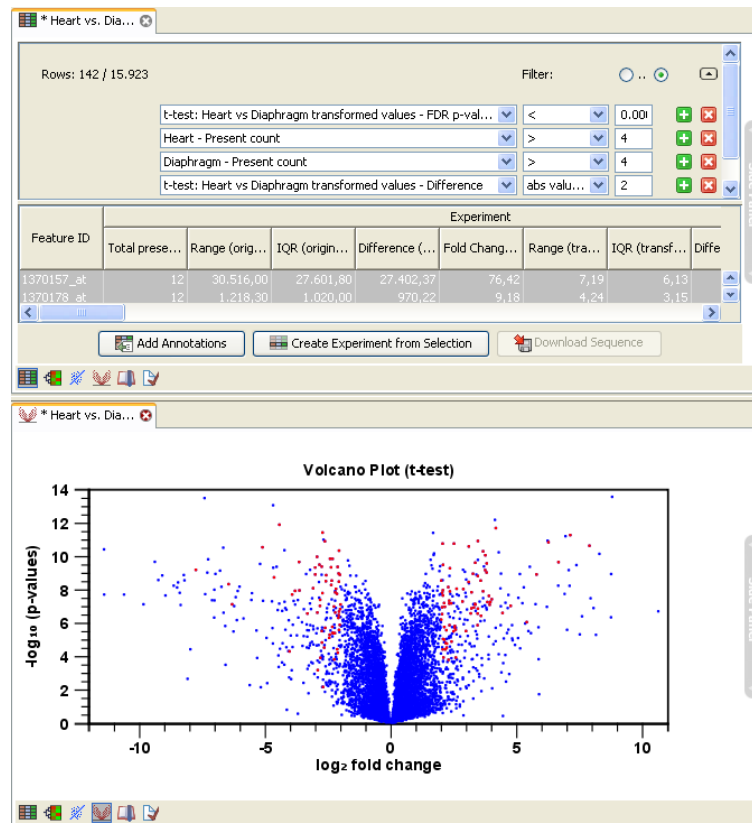
Figure 2.39: *142 genes out of 15923 selected.*

2.7.5   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.

2.8   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).

2.8.1   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** | **Transcriptomics Analysis** (⬚)| **Annotation Test** | **Add Annotations** (⬚)

Select the experiment created in the previous tutorial and the annotation file (⬚) and click **Next** and **Finish**.

2.8.2   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.

2.8.3   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** | **Transcriptomics 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 2.40).

Click **Next** and **Finish** to perform the test. The result is shown in figure 2.41.

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.

2.8.4   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

Figure 2.40: *Testing on GO biological process.*

Figure 2.41: *The result of testing on GO biological process.*

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 | Transcriptomics 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 2.42.

Click **Next** and select the **Transformed expression values**. Click **Finish**. The result is shown in figure 2.43.
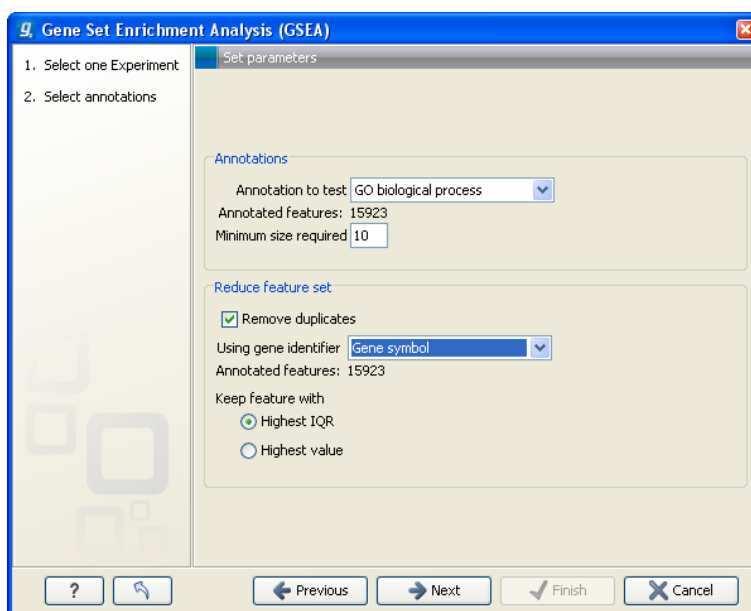
Figure 2.42: *Gene set enrichment analsysis based on GO biological process.*

Figure 2.43: *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.

2.9 Tutorial: Phylogenetic Trees

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 2.44).
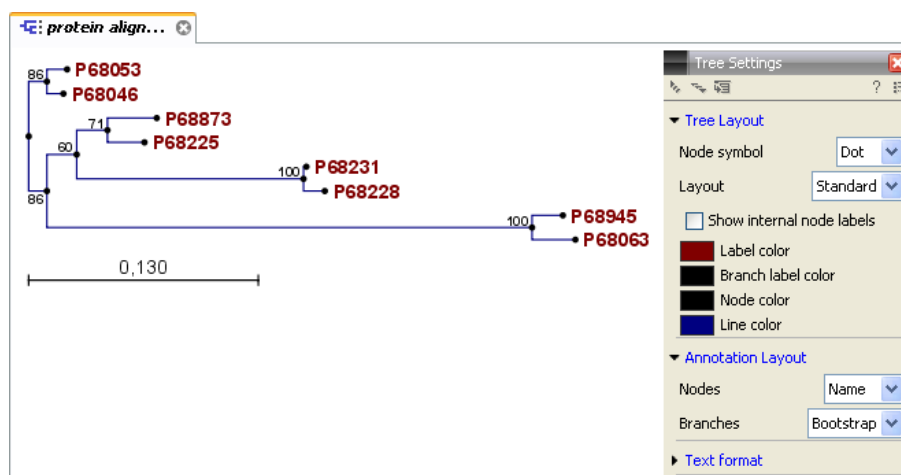
Figure 2.44: *After choosing which algorithm should be used, the tree appears in the View Area. The Side panel in the right side of the view allows you to adjust the way the tree is displayed.*

2.9.1   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.

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.

Chapter 3

Introduction

The CLC Phylogeny Module is a collection of tools for phylogenetic analysis. When installed, the standard viewer for phylogenetic trees is replaced with a new and more advanced viewer. This new phylogenetic tree viewer provides many new features for customizing tree visualization such as:

- Circular and radial layouts.

- Adjustment of node size and branch width.

- Curved branches.

- Grouping of nodes and visualization of these.

- Visualization of metadata.

The plugin also gives access to two new tools in:

Toolbox | Classical Sequence Analysis ( ) | Alignments and Trees ( )

The first of these tools (**Model Testing**) is a tool for testing substitution models for maximum likelihood reconstruction of trees. The other tool (**K-mer Based Tree Construction**) can reconstruct trees using a K-mer based approach, which does not require a multiple alignment of the sequences. For more information on these tools, please see http://clcsupport.com/phylogenymodule/current/User_Manual.pdf.

One of the most interesting features in this plugin is the visualization of metadata, which provide easy and flexible ways to visualize different types of metadata.

This tutorial will briefly introduce reconstruction of phylogenetic trees and visualization of these with the new tree viewer. The main focus in this tutorial is the visualization of metadata as a powerful tool for analyzing your data.

3.1 Overview

The features demonstrated in this tutorial include:

- Alignment of sequences.

- Reconstruction of phylogenetic trees.

- Introduction to the new tree viewer.

- Metadata and visualization of these.

- Grouping of nodes.

- Labeling of subtrees.

3.2 Pre-requisites

To complete this tutorial, you must be working with the *CLC Main Workbench* version 6.9+ or the *CLC Genomics Workbench* version 6.5+. Minimum recommended machine specifications can be found at http://www.clcbio.com/support/system-requirements/.

3.3 Example Dataset

For this tutorial we will be using a dataset containing sequences and metadata from the Viral hemorrhagic septicemia virus (VHSV). The VHSV is a fish novirhabdovirus (negative stranded RNA virus) with an unusual broad host spectra: It has been isolated from more than 80 different fish species around the Northern hemisphere. The dataset contains two files. One file contains sequences encoding the viral surface glycoprotein of VHSV in fasta format. The other file contains key metadata information in an excel spreadsheet.

Sequence data:
http://download.clcbio.com/testdata/phylogeny_module_tutorial_data/Phylogeny_
module_example_data.fa
Metadata:
http://download.clcbio.com/testdata/phylogeny_module_tutorial_data/Phylogeny_
module_tutorial_meta_data.xls

The metadata file contains the following categories of metadata, which will be used later in this tutorial.

- **Sequence** - The nucleotide sequence of a surface glycoprotein.

- **Strain** - The virus strain.

- **Host** - The fish from which the sample was obtained.

- **Water** - The type of water the fish lives in.

- **Country** - The country where the fish was caught.

- **ACCNo** - The accession number of the virus genome.

- **Year** - The year in which the sample was obtained.

3.4   Importing the Sequence Data

To download the example data, click the two links above. The sequence data can be imported into the CLC Workbench by following these steps:

1. Start the CLC Workbench.

2. To import the sequence data go to the toolbar:

    Import ( ) | Standard Import ( )

3. Choose the file "phylogeny_module_example_data.fa". Ensure that the import type under **Options** is set to **Automatic import** and click **Next** (figure 3.1).

4. Select the location where you want to store the imported sequences and click **Finish**.
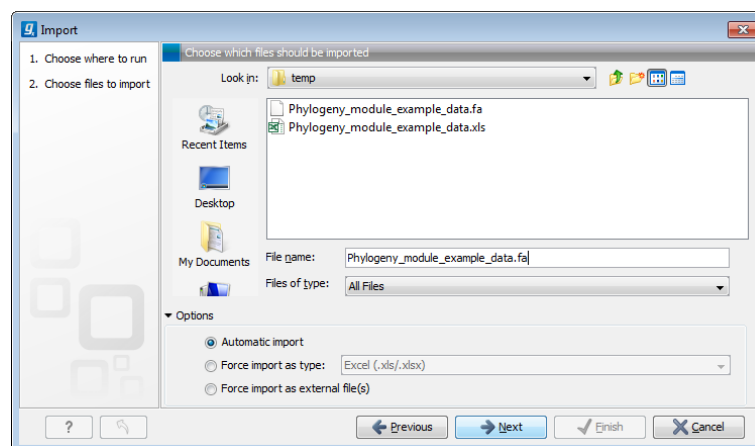
Figure 3.1: *Import of the virus sequence data.*

Chapter 4

Aligning Sequences

Most phylogenetic reconstruction methods require a multiple alignment of the input sequences to reconstruct the corresponding phylogenetic tree. You can create an alignment of the imported sequences by following these steps:

1. Start the **Create alignment** tool:

   Toolbox | Classical Sequence Analysis (🖼) | Alignments and Trees (📁) | Create Alignment (▦)

2. Select the imported sequences **Phylogeny_module_example_data** and click on the button labeled **Next**.

3. Use the default settings in step 3 of the wizard but ensure that the **Less accurate (fast)** option is selected (figure 4.1). Click on the button labeled **Next**.

4. Choose the option **Save** and select the location where you wish to store the alignment. Click on the button labeled **Finish**.
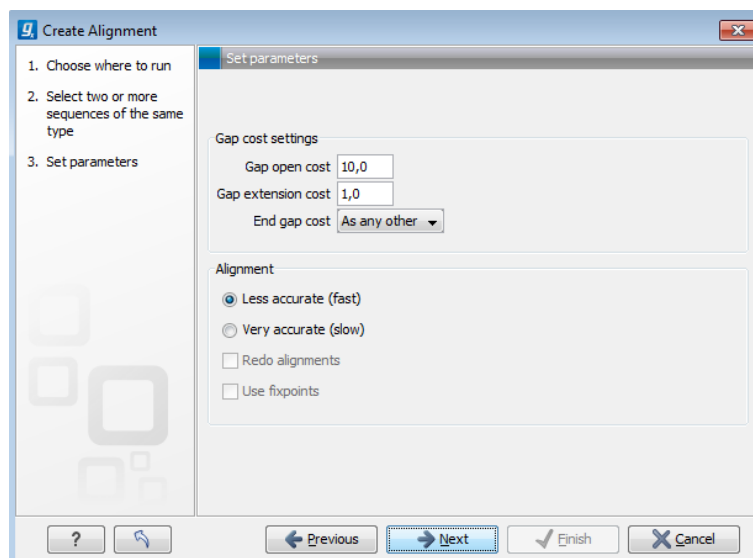
Figure 4.1: *Creating a multiple alignment of the sequence data.*

Note! It is possible to reconstruct a phylogenetic tree without first creating a multiple alignment of the input sequences. The **K-mer Based Tree Construction** tool utilizes a K-mer based approach to estimate pair-wise distances between the input sequences. This distance estimate can then be used to reconstruct the tree using either the UPGMA or Neighbor-Joining algorithms. If the input dataset is large and/or contain very long sequences, it can be an advantage to use this tool for reconstructing trees to avoid the time consuming task of creating a multiple alignment.

Chapter 5

Reconstructing the Tree

Using the multiple sequence alignment created in the previous step, the phylogenetic tree can now be reconstructed using one of the following two tools:

- **Create Tree** - Constructs trees using one of two distance based methods:
  - UPGMA
  - Neighbor-Joining

- **Maximum Likelihood Phylogeny** - Reconstructs phylogenetic trees using a maximum likelihood approach.

In this tutorial we will use the Neighbour-Joining method which is a fast and fairly accurate method for phylogenetic reconstruction.

1. Start the **Create Tree** tool:

   Toolbox | **Classical Sequence Analysis** (🖼) | **Alignments and Trees** (📋) | **Create Tree** (📲)

2. Select the multiple alignment.

3. Leave all options to the default values and save the tree (figure 5.1).

It takes some time to reconstruct the tree as the default parameters include bootstrapping with 100 replicates. This basically means that the tree will be reconstructed 101 times to test the confidence of each internal node in the resulting tree.
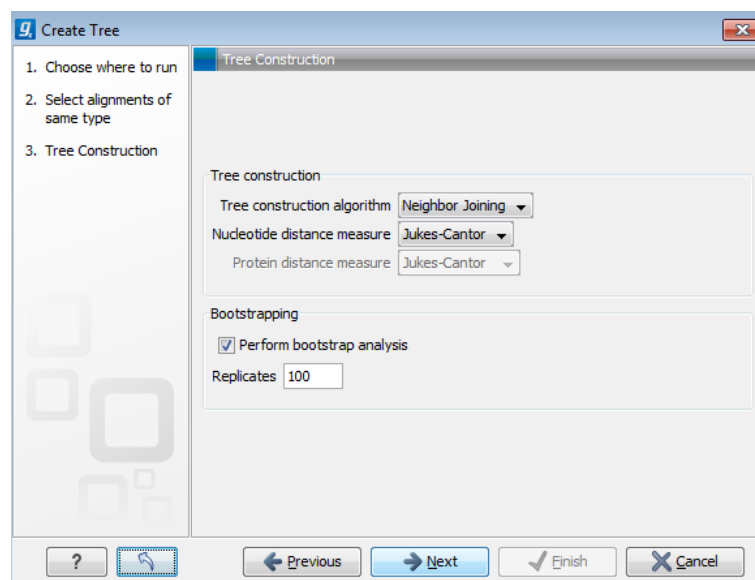
Figure 5.1: *Reconstruct a neighbor-joining tree with 100 bootstrap replicates.*

Chapter 6

Visualizing the Tree

To open a tree in the new tree viewer, double click the tree object (figure 6.1) in the **Navigation Area**. The viewer can display the tree in five different layouts:
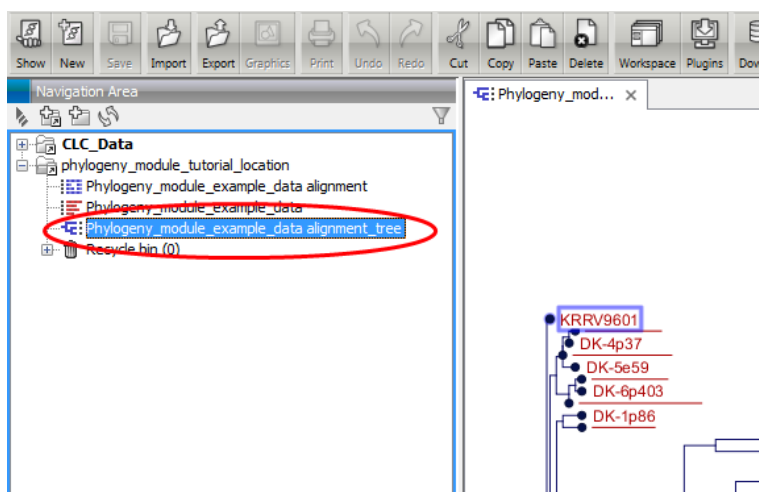
Figure 6.1: *Double click the tree object (in the red circle) to open the tree in the View Area.*

- **Phylogram** - The tree is displayed as a rooted tree where branch lengths correspond to the computed lengths.

- **Cladogram** - The tree is displayed as a rooted tree where branch lengths are ignored and all leaves are aligned to the right.

- **Circular Phylogram** - Same as a Phylogram but with the leaves in a circular layout.

- **Circular Cladogram** - Same as a Cladogram but with the leaves in a circular layout.

- **Radial** - The tree is displayed as an unrooted tree where branch lengths correspond to the computed lengths.

In this tutorial we will primarily use the Phylogram layout but we encourage you to try the different layouts while going through this tutorial.

Zooming and scrolling work in the same way as other viewers/editors in the Workbench. When zooming in on a large tree, one can easily loose orientation. Hence, in order to reduce the need

75

for zooming out again, the new tree viewer includes a small minimap in the right side panel (figure 6.2). The minimap shows the full tree, with a grey rectangle highlighting the area of the tree that currently is shown in the View Area. The grey rectangle can be dragged around in the minimap with the mouse, and as the minimap and the tree in the View Area are synchronized, the minimap can be used to keep the orientation. Try to click on the minimap and drag the grey rectangle around to see how the tree in the View Area follow the movements.
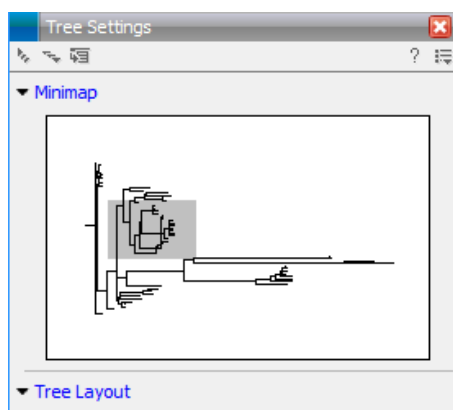
Figure 6.2: *The minimap. The grey rectangle represents the area of the tree that is visible in the View Area.*

Similar to the standard viewer for phylogenetic trees, the right side panel have different options for modifying the visualization. The new tree viewer have some new options compared to the old viewer such as adjustment of tree colors, node shape, node size and the shape of branches. For more details on these options, please refer to the manual for the CLC Phylogeny Module (see http://clcsupport.com/phylogenymodule/current/User_Manual.pdf).

6.1 Bootstrap Values

In the right side panel under **Bootstrap settings** enable the option **Show bootstrap values** (figure 6.3). Now each internal node is labeled with a value between 0 and 100. These values represent confidence levels where a high confidence indicate a clade, which is strongly supported by the data from which the tree was reconstructed. Bootstrap values are useful for identifying clades in the tree where the topology (and branch lengths) should not be trusted.
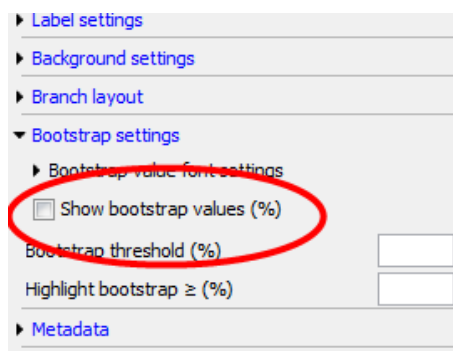
Figure 6.3: *Enabling bootstrap values.*

To help get an overview of the confidence values we can highlight all branches that lead to a node with a bootstrap value of e.g. $\geq 95\%$. To do this enter "95" in the field labeled **Highlight**

bootstrap above under **Bootstrap settings** in the right side panel (figure 6.4). Figure 6.5 shows an example of this.
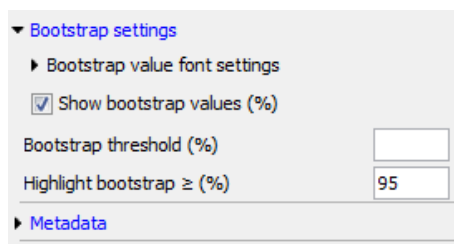
Figure 6.4: *Highlighting edges with bootstrap values equal to or larger than a threshold.*
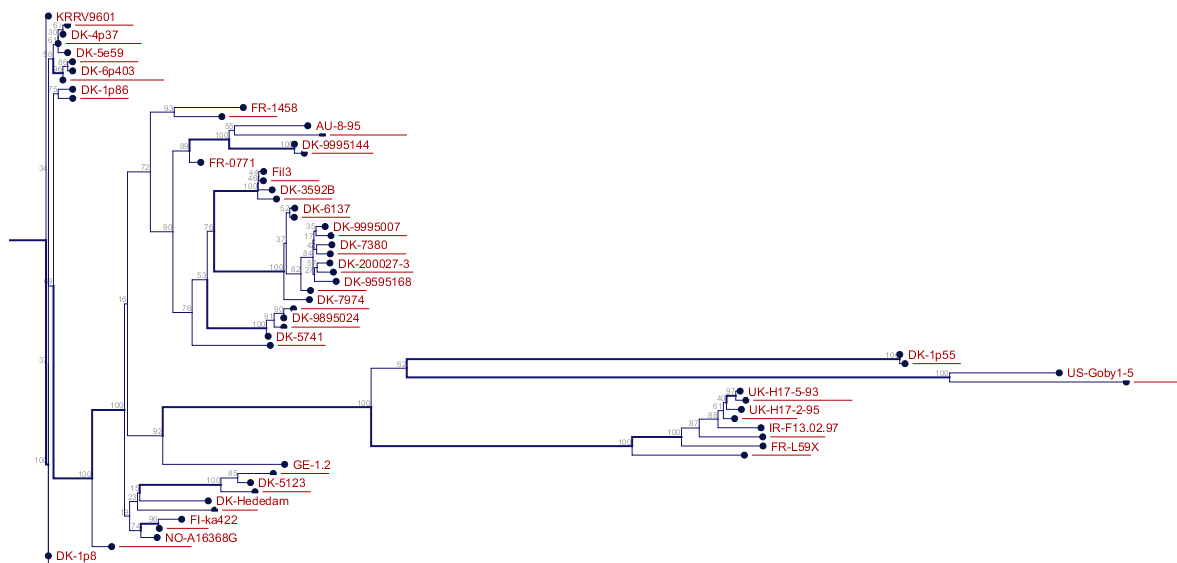
Figure 6.5: *The tree where edges corresponding to nodes with bootstrap values ≥95% are highlighted.*

Another way to visualize bootstrap values is to collapse internal nodes with bootstrap values under a certain threshold. After removing a node the child nodes will be connected to the parent of the collapsed node which creates a multifurcating tree containing only high confidence nodes. To enable this visualization enter a threshold (e.g. 95%) in the field labeled **Bootstrap threshold** under **Bootstrap settings** in the right side panel. Figure 6.6 shows an example of this.

6.2   Metadata

When a tree is reconstructed, some mandatory metadata will be added to nodes in the tree. These metadata are special in the sense that the tree viewer has specialized features for visualizing the data and some of them cannot be edited. Examples of mandatory metadata:

- **Name** - The node name.

- **Branch length** - The length of the branch which connects a node to the parent node.

- **Bootstrap value** - The bootstrap value for internal nodes.

- **Size** - The length of the sequence which corresponds each leaf node. This only applies to leaf nodes.
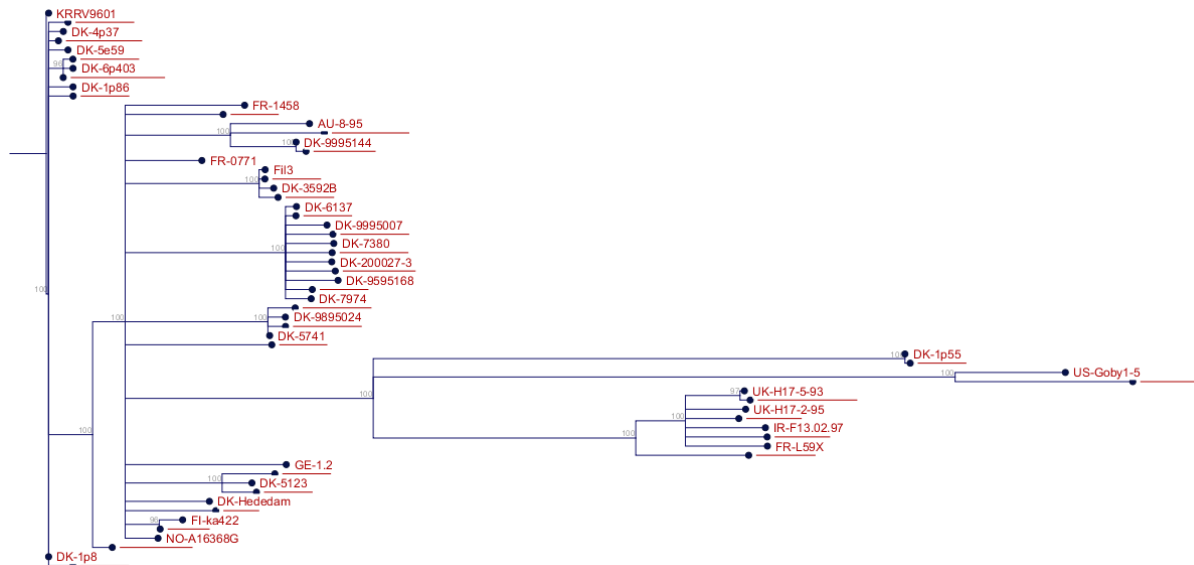
Figure 6.6: *The tree where edges corresponding to nodes with bootstrap values <95% are collapsed.*
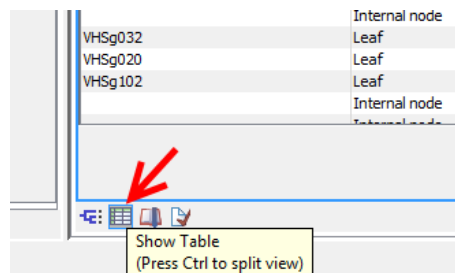
Figure 6.7: *The metadata table button.*

- **Start of sequence** - The first 50bp of the sequence corresponding to each leaf node.

To view a table of the metadata associated with a tree click the **Show Table** button (figure 6.7). This opens a table where each row represents a node and the columns contain different categories of metadata. To open the metadata table in a split view where both the tree and the table is visible at the same time, click the **Show Table** while holding down Ctrl (or (⌘) for Mac) (figure 6.8).

6.3 Importing Metadata

Additional metadata can be imported by clicking the **Import Metadata** button below the metadata table (figure 6.8). To import the metadata contained in the file "Phylogeny_module_example_meta_data.xls" follow these steps:

1. Click the **Import Medata** button.

2. Click on the folder icon (📁) (next to the top text field) and select the file "Phylogeny_module_example_meta_data.xls".

3. Use the default settings and click **Finish**
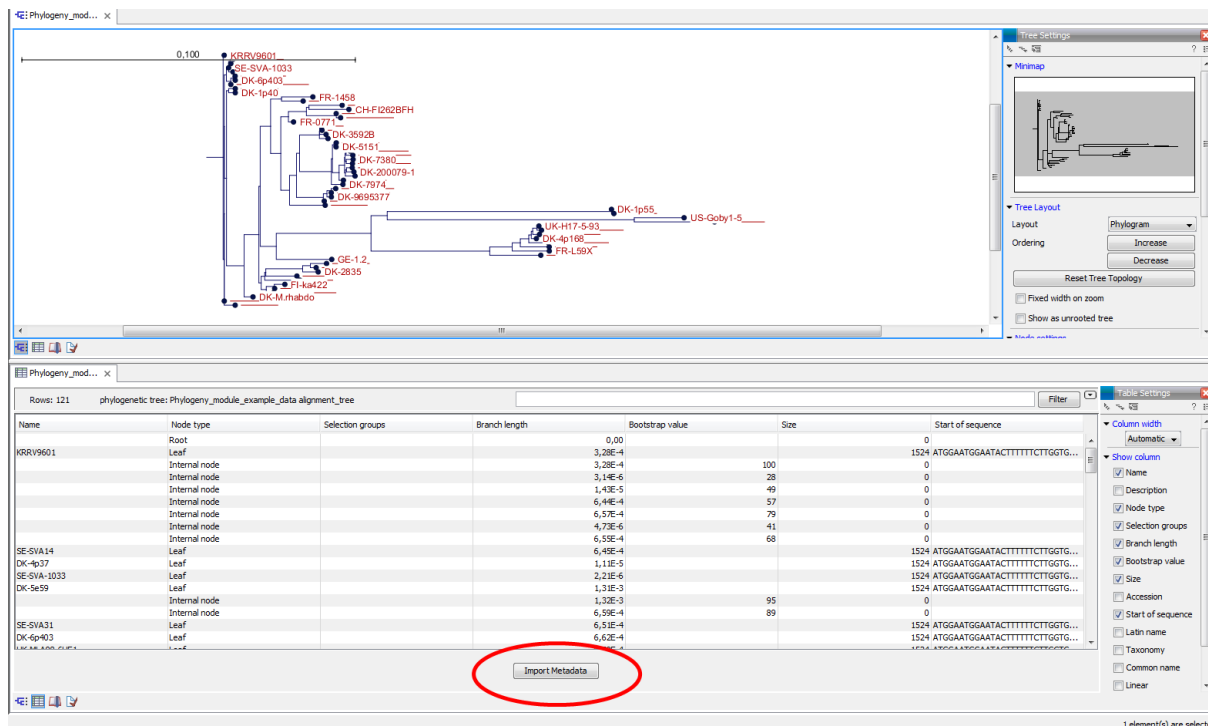
Figure 6.8: *Metadata table and tree viewer in split view mode. The red circle indicate the button for importing additional metadata.*
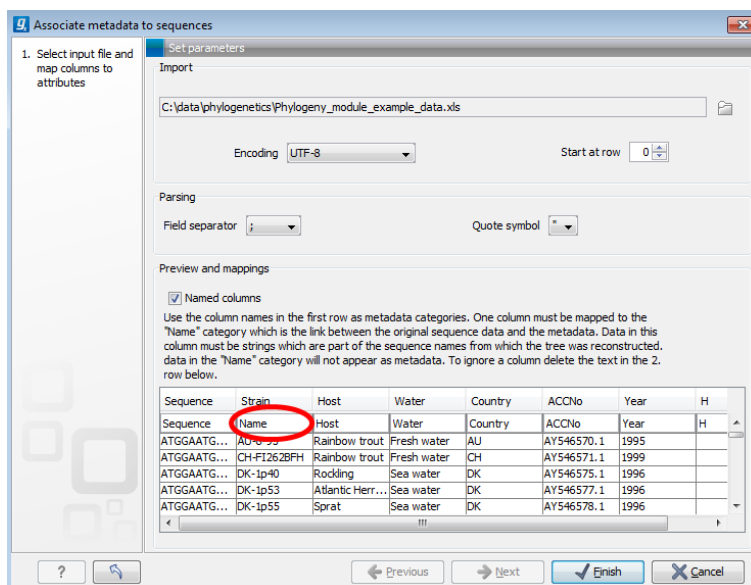
Figure 6.9: *The metadata import wizard. In this example the "Strain" column is mapped to the "Name" category and hereby used to map metadata to the tree nodes.*

The file "Phylogeny_module_example_meta_data.xls" is an Excel file containing 7 categories of metadata (see section 3.3). The importer uses node names as an identifier for linking each row of metadata to nodes. For this to work, a column in the imported data must contain node names or unique parts of these. The metadata import wizard allows the user to select the column that contains node names using a preview table (figure 6.9). By changing the target metadata category of a column (second row in the preview table) to the special **Name** category, data in that column will be used to map metadata to nodes. The importer will usually be able to guess the

correct column but in some cases where the metadata only contain parts of the name, it can be useful to manually set the **Name** column.

Figure 6.10: *Content of the metadata table after additional metadata has been imported.*

After the import is complete, a number of new metadata columns have appeared in the metadata table (figure 6.10).

A quick way to see the metadata for a specific node in the tree is to hold the mouse over the node for a few seconds until a tooltip appears (figure 6.11).
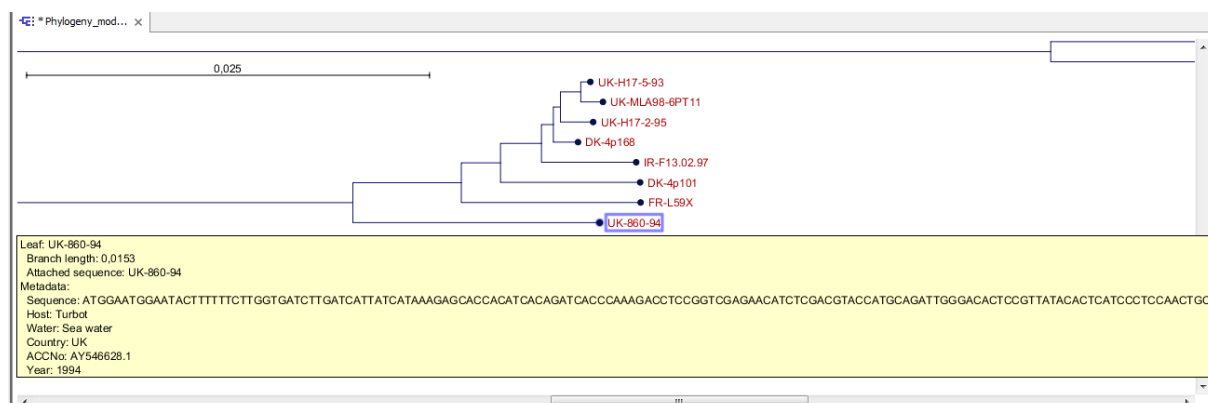
Figure 6.11: *When holding the mouse over a node in the tree a tooltip shows all metadata for that node.*

6.4   Visualization of Metadata

Visualization of metadata is done through the options under **Metadata** in the right side panel (see figure 6.12).

The tree viewer has several options for visualizing both textual and numeric metadata. This tutorial will demonstrate three options and for a description of the other visualization options offered by the tree viewer, please refer to the CLC Phylogeny Module manual (`http://clcsupport.com/phylogenymodule/current/User_Manual.pdf`).

Node color:

1. Under **Metadata** in the right side panel click **Node color**.

2. In the dropdown menu select **Host** (figure 6.13).
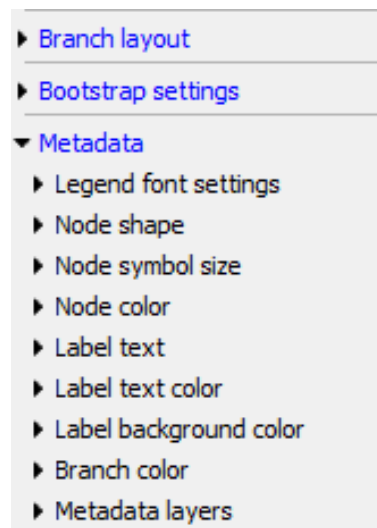
Figure 6.12: *The metadata visualization options in the right side panel of the tree viewer.*
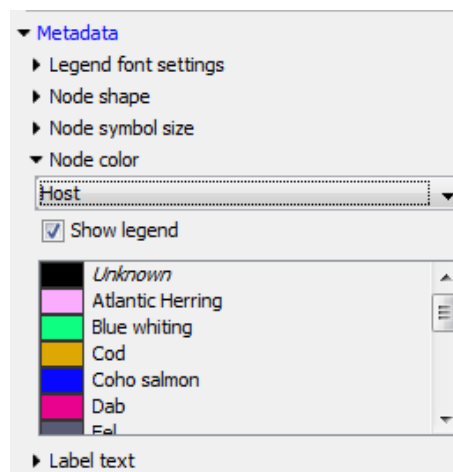
Figure 6.13: *Color nodes by the host which the sample was extracted from.*

Nodes are now colored according to the host organism from which the virus sample was extracted. Random colors are automatically assigned to each entry in the metadata category. To change a color go to **Node color** in the right side panel, left click on the color you want to change and choose a new color in the color chooser that pops up (figure 6.14). If an entry does not contain any data, the corresponding node is assigned a default color. This default color is the first color in the legend and has the label **Unknown**. The default color can be adjusted just like any other color.

Node symbol size:

1. Under **Metadata** in the right side panel click **Node symbol size**.

2. In the dropdown menu select **Water** (figure 6.15).

All nodes are now assigned sizes which symbolize the water type in which the host fish lives. Each type of water is automatically mapped to a node size, and these sizes can be changed by using the sliders that are now visible under **Node symbol size** (figure 6.15).

Metadata layers: The metadata can be visualized on the tree as color-code layers.
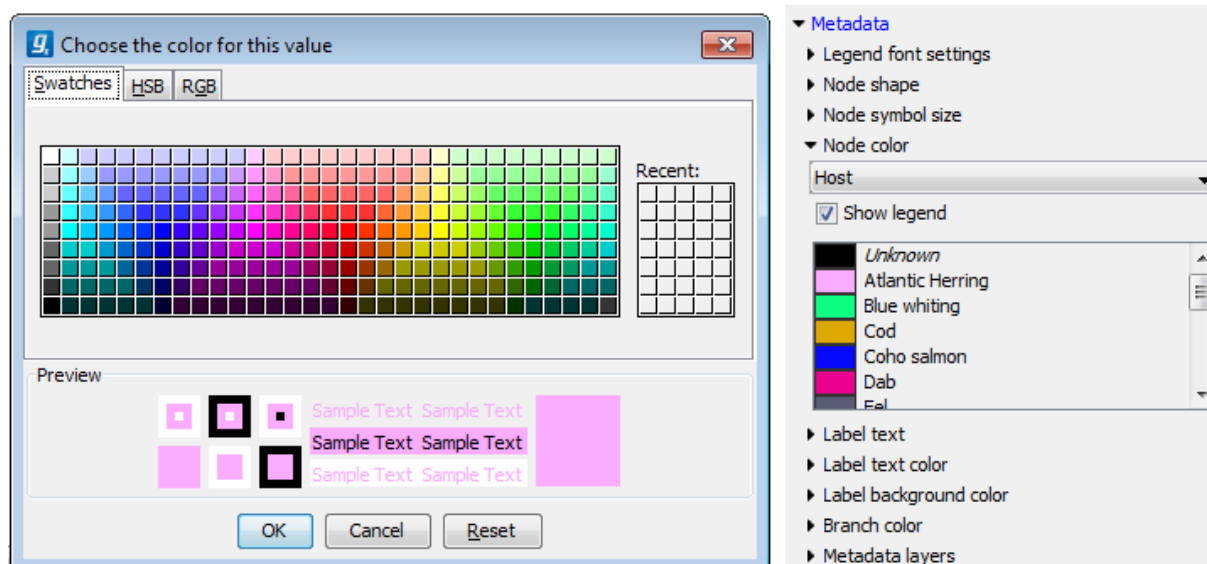
Figure 6.14: *The color chooser can be used to select new colors for each metadata category.*
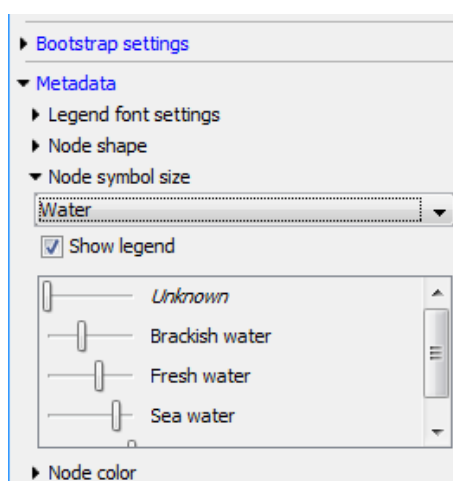
Figure 6.15: *Visualization of the water type in which the host fish lives using node sizes.*

1. Under **Metadata** in the right side panel first select **Metadata layers** and then select **Metadata layer #1**.

2. In the dropdown menu select **Country** (figure 6.16).

The countries where each sample was obtained are now visualized by colors in the color-code layer at the bottom of the tree (figure 6.17). Similar to the node color visualization, the color for each entry in the "Country" category can be changed under **Metadata layer #1**. It is also possible to adjust the thickness of the color-code layer by using the slider labeled **Layer thickness** under **Metadata layer #1**. If the **Show layer names** option is enabled, there is a limit to how thin a layer can be made. This is to ensure that the name of different color-code layers do not overlap. By disabling the **Show layer names** option, layers can be made as thin as a single pixel.

After adding the color-code layer, a new option has appeared under **Metadata layers** in the right side panel called **Metadata layer #2**. By selecting a metadata category for this option, a new color-code layer will be added on top of the first color-code layer. By visualising metadata using multiple color-code layers, users can get a quick overview of data in different metadata categories
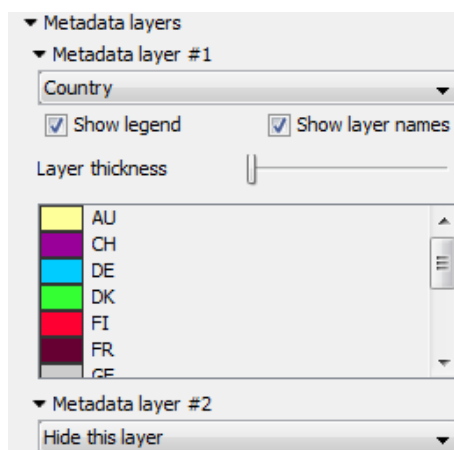
Figure 6.16: *Visualization of the country in which the host fish was caught using a color-code layer.*
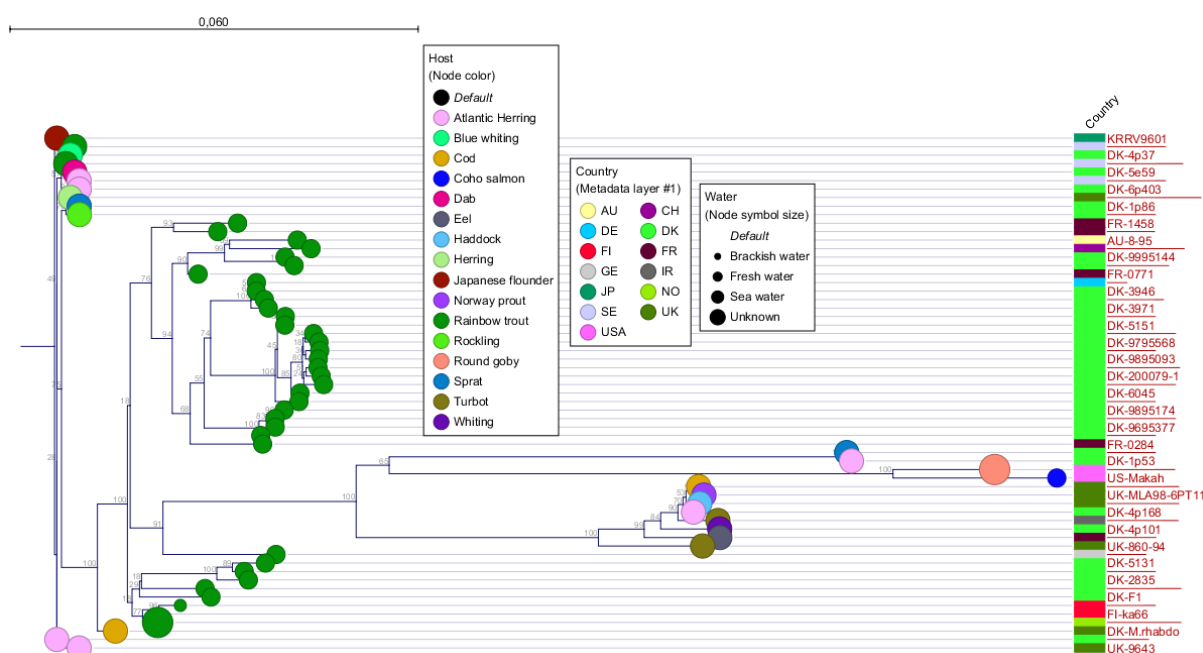
Figure 6.17: *Tree where water type, host and country are visualized.*

which makes it possible to observe complex correlations. Figure 6.18 shows an example where three metadata categories are visualized using color-code layers.

6.5 Create and Modify Metadata

Metadata do not have to imported from an external file. It is possible to both create and modify metadata through the metadata table and the tree viewer. A new metadata category can be created using the tree view by following these steps:

- Select one or more nodes in the tree view

- Right click one of the selected nodes and select **Assing metadata** (figure 6.19).

- Type in a name for metadata category, and use the **Value** field to assign a value to all selected nodes.
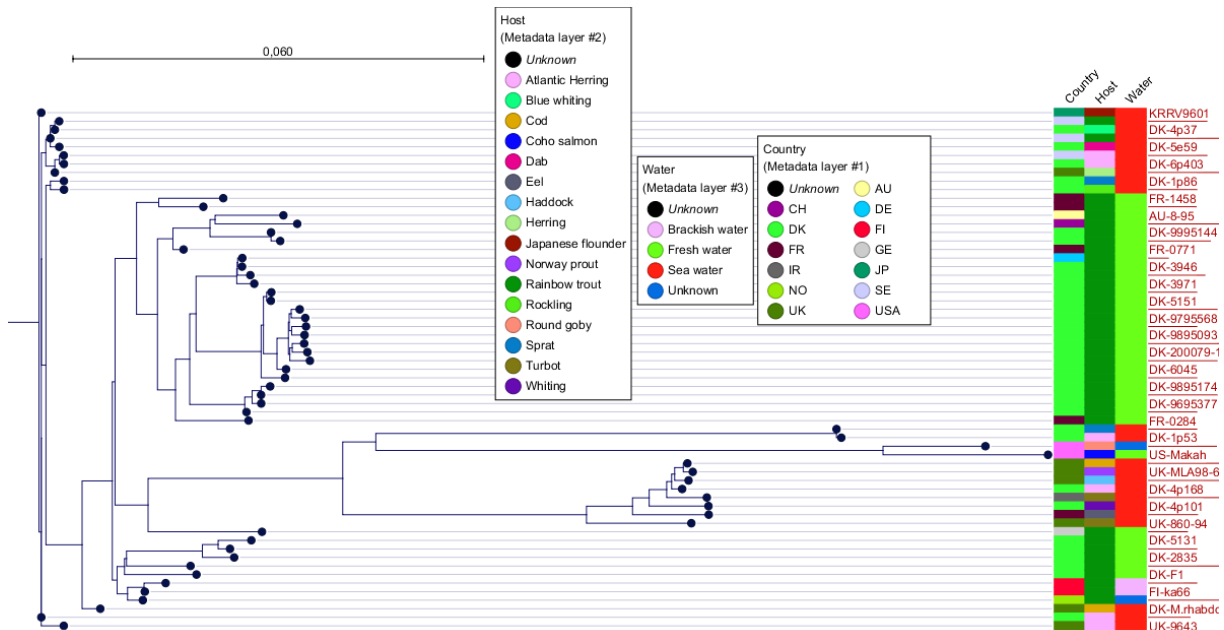
Figure 6.18: *Tree where water type, host and country are all visualized with color-code layers.*

- Click **Add**
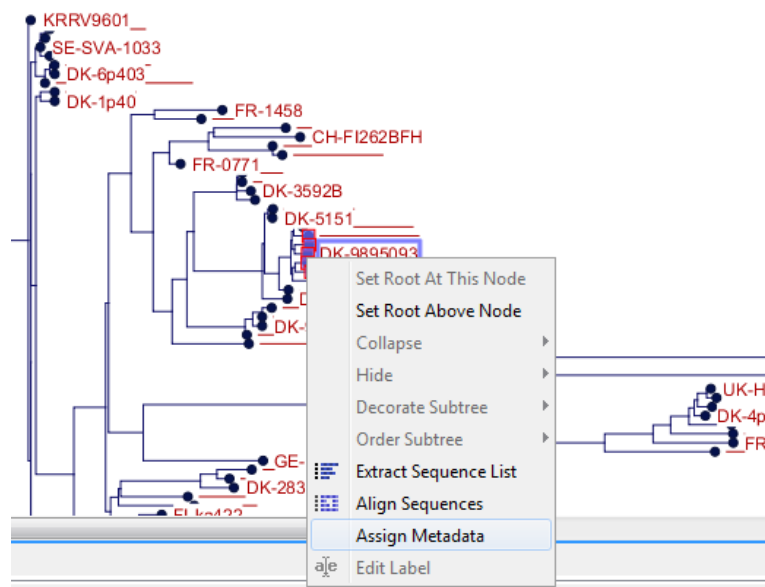
Figure 6.19: *Creating a new metadata category using the right click context menu.*

Now a new metadata category has been created with values assigned to all selected nodes. To assign values in the same metadata category to other nodes, follow the steps above but instead of writing a new name for a metadata category, use the drop down menu in the **Name** field to select the newly created category (figure 6.20). The value entered in the **Value** field will now be assigned to the selected category for all selected nodes.

Manipulation of metadata through the metadata table is very similar to how it is done through the tree view. The only difference is that one has to select one or more rows in the table instead of selecting nodes in the tree.
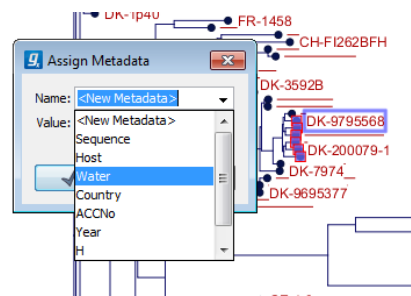
Figure 6.20: *Modifying an existing metadata category.*
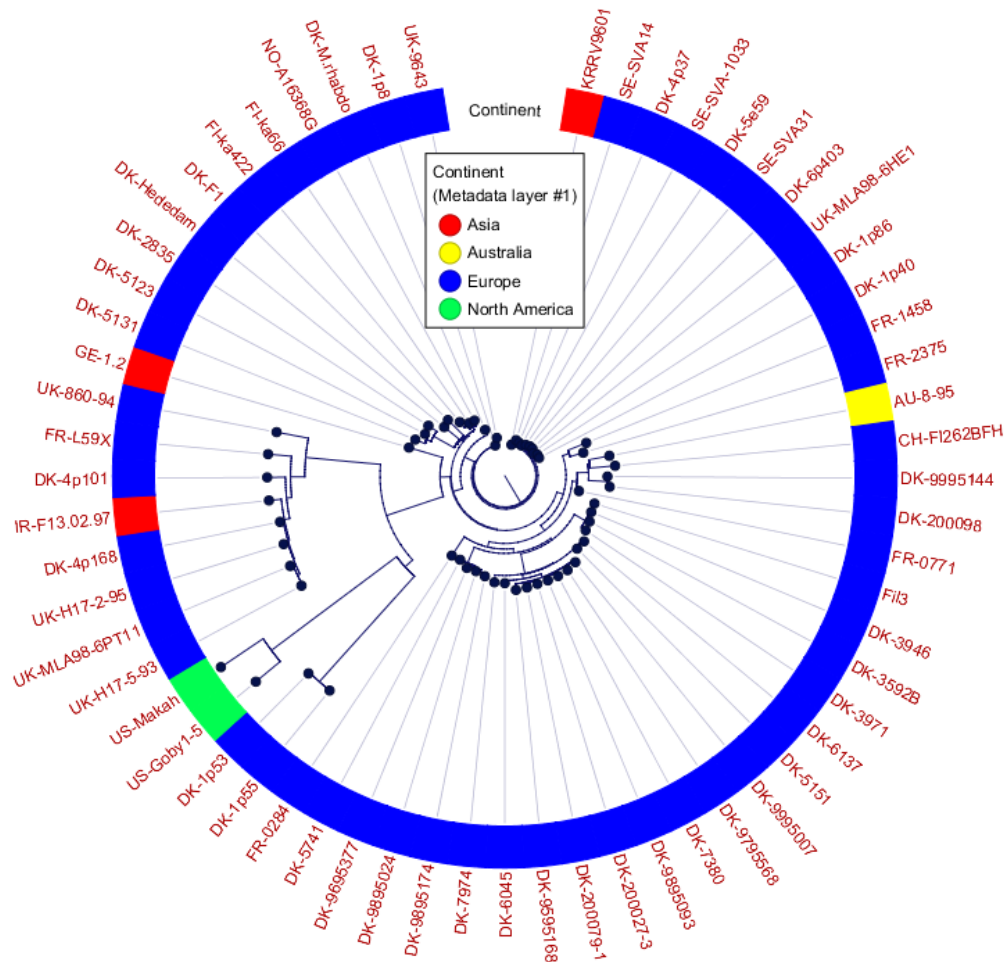
Figure 6.21: *Visualization of the continents where the host fish was found. The circular phylogram layout is used here.*

Manually created metadata categories can be visualized in the same way as any other metadata category. An example where a new metadata category called "Continents" has been created and where all leaf nodes have been assigned values in that category according to the continent where the sample was obtained, is shown in figure 6.21.

Chapter 7

Subtree Labels

It is often convenient to put labels on subtrees to illustrate what the nodes in a subtree represents. This can be done in the following way:

1. Right click an inner node in the tree.

2. Select **Decorate Subtree** and then select **Set Subtree Label** (figure 7.1).

3. Enter a label text and select the line color.

The result is a labelled line at the bottom of a subtree where the inner node which was clicked in step 1 is the root. Figure 7.2 shows an example where different subtrees have been labeled.
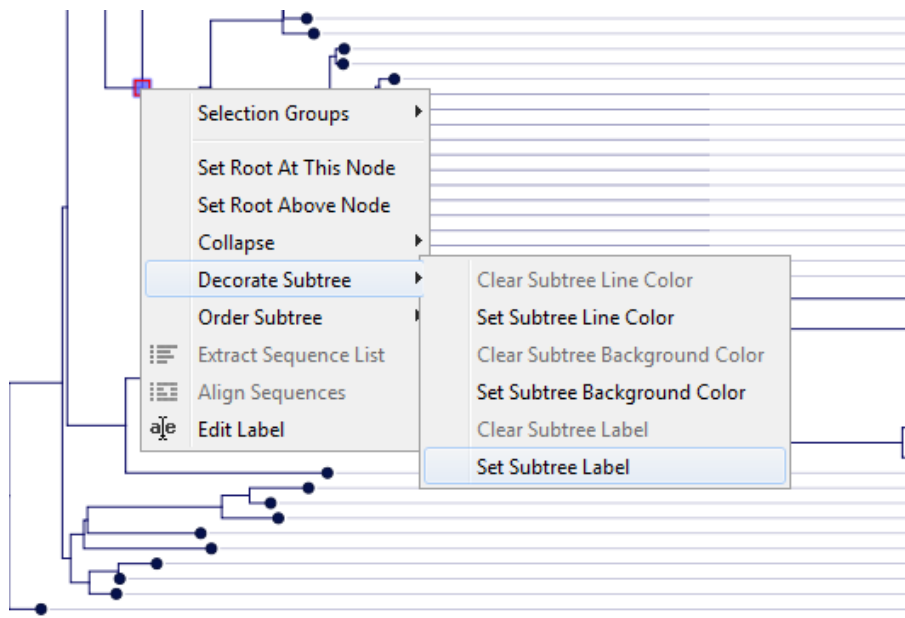
Figure 7.1: *Creating a label for a subtree.*
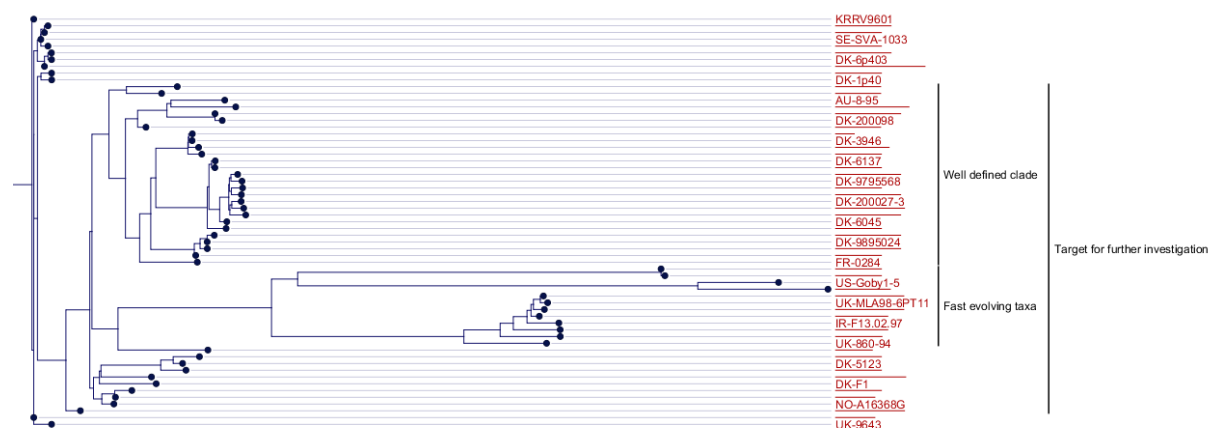
Figure 7.2: *A tree with three labeled subtrees. One label is for a subtree in which both the other two labeled subtrees are contained.*

7.1 Tutorial: Assemble to Reference

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 using the *CLC Main Workbench*. 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" folder in the "Sequencing data" folder of the **Example data** in the **Navigation Area**. If you do not have the example data, please go to the **Help** menu to import it.

7.1.1 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 | Sequencing Data Analysis (⚛)| 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 7.3).

There is no vector contamination in these data, so 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 are actually annotated, not removed (see figure 7.4). By choosing **Save**, the Trim annotations will be saved directly to the sequences, without opening them for you to view first.

These annotated parts of the sequences will be ignored in the subsequent assembly.
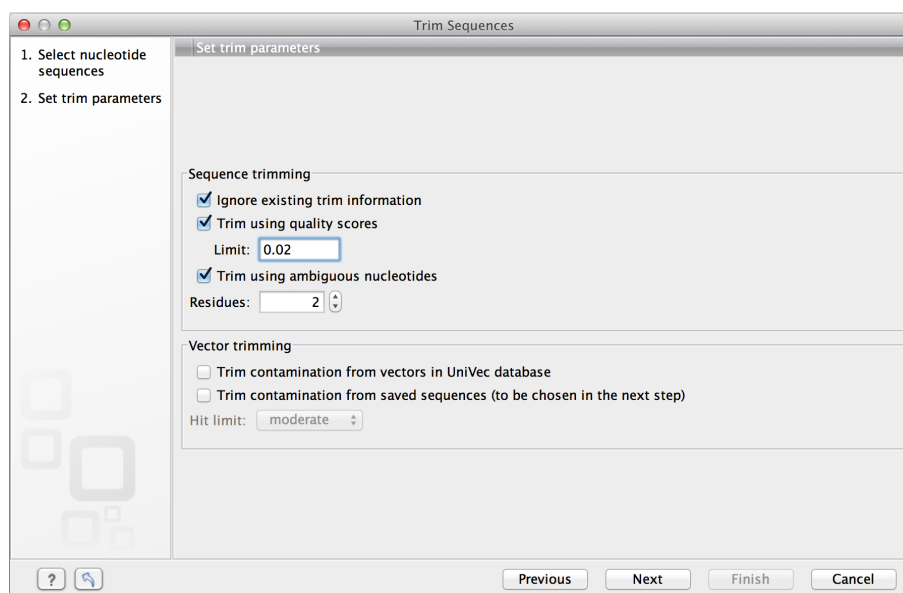
Figure 7.3: *Specifying how sequences should be trimmed. A stringent trimming of 0.02 is used in this example.*
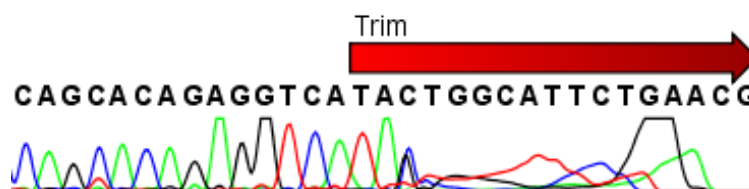
Figure 7.4: *Trimming creates annotations on the regions that will be ignored in the assembly process.*

A natural question 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.

7.1.2 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 | Sequencing Data Analysis (⚇)| 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 7.5). You can leave the other options in this window set to their defaults.
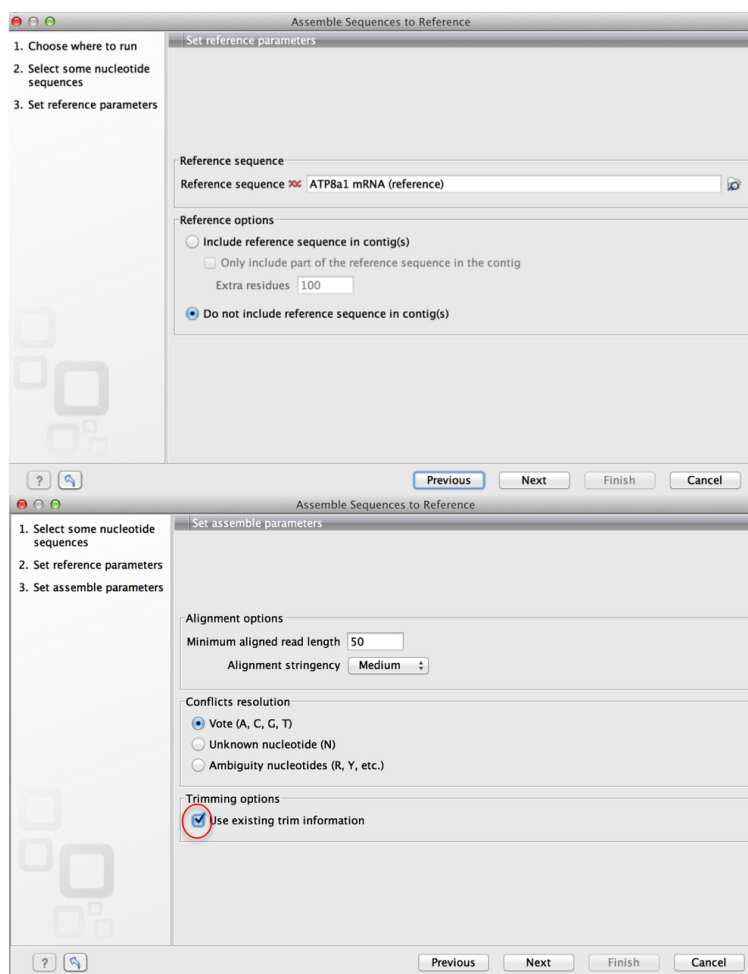
Figure 7.5: *The "ATP8a1 mRNA (reference)" sequence selected as reference sequence for the assembly.*

Click **Next** and choose to **Use existing trim information** (that you have just added).

Click **Finish** and the assembly process will begins.

7.1.3 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 7.6):

> **Alignment info in Side Panel** | **Coverage** | **Graph**

This overview can be an aid in determining whether coverage is satisfactory, and if not, which regions a new sequencing effort should focus on. Next, we go into the details of the contig.

7.1.4 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; you can also use ',' and '.' keys to move
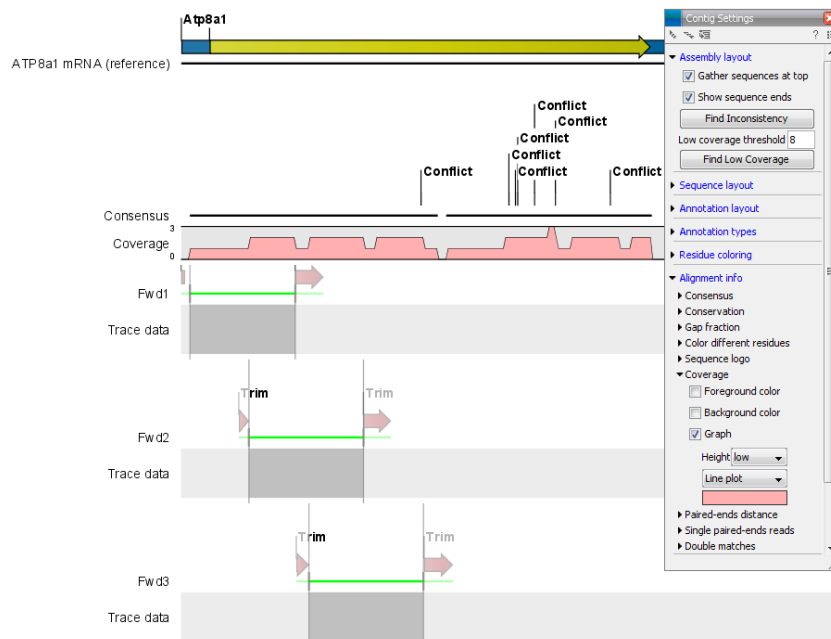
Figure 7.6: *An overview of the contig with the coverage graph.*

back and forth between conflicts (see figure 7.7).
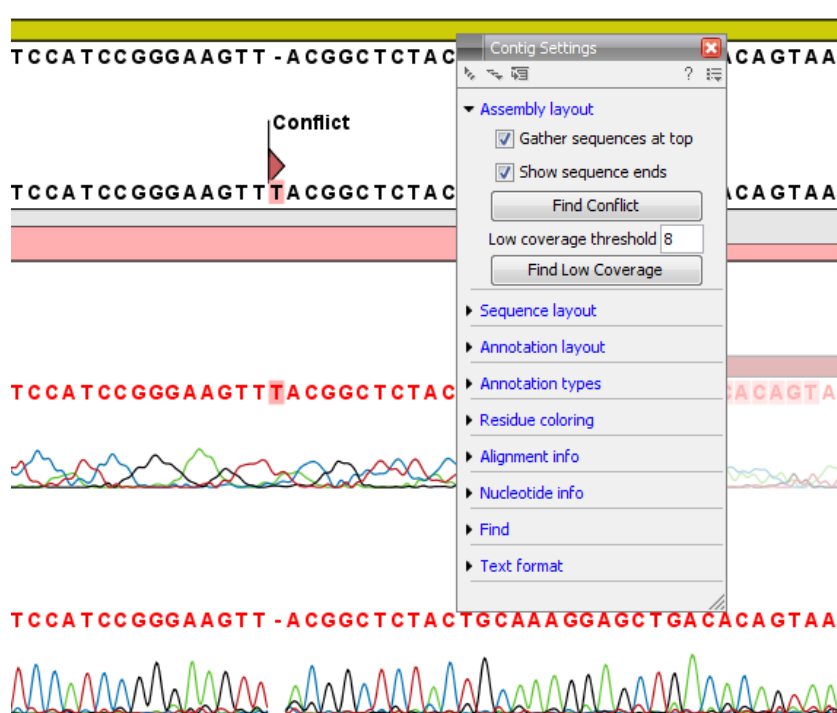
Figure 7.7: *Using the Find Conflict button highlights conflicts.*

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

should trust the "Rev2" read. In addition, you can see that we are close to 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 by placing the cursor to the left of it and dragging the cursor across the T. Press **Delete** (🗑).

This will resolve the conflict.

7.1.5   Including regions that have been trimmed off

Clicking the **Find Conflict** button again will find the next conflict.

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 actually seem to be fine, so we could just as well include this information in the contig.

You can see where the trimmed region begins. To include part of the trimmed region of Fwd2 in the contig, move the vertical slider at position 2073 to the left (see figure 26.21).
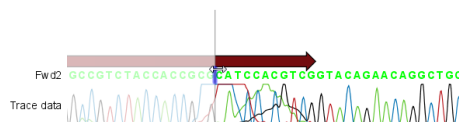
Figure 7.8: *Dragging the edge of the trimmed region.*

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.

7.1.6   Inspecting the traces

Clicking the **Find Conflict** button again will find the next conflict.

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 using the Tool Bar zoom function (🔍) | Click the selected base | Click again three times**

Now you have zoomed in on the trace (see figure 7.9).

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 26.2).

7.1.7   Synonymous substitutions?

In this case we have sequenced the coding part of a gene. Often you want to know what 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**
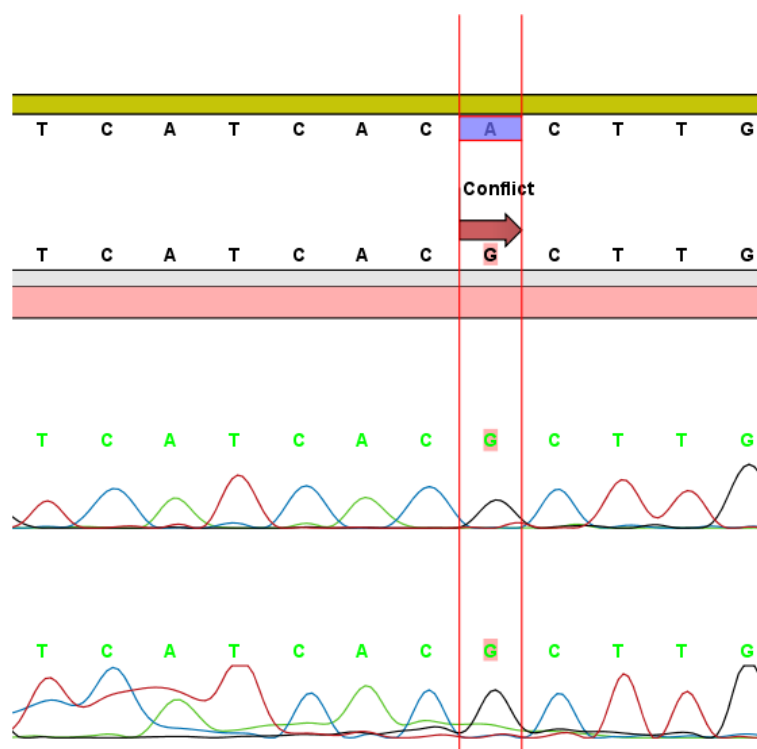
Figure 7.9: *Now you can see all the details of the traces.*
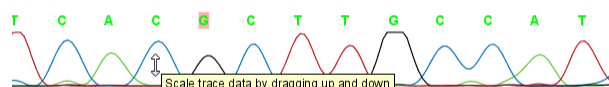
Figure 7.10: *Grab the traces to scale.*

The result is shown in figure 7.11.

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 orange. If it was a non-synonymous substitution, it would be colored in red.

7.1.8   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.

Figure 7.11: *Showing the translation along the contig.*

7.1.9   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 **History** (🔘) icon at the bottom of the view.

In the history, you can see the details of each change (see figure 7.12).
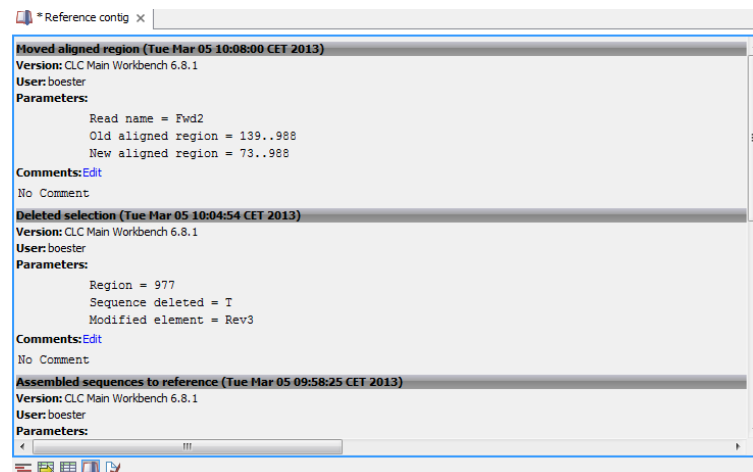
Figure 7.12: *The history of the contig showing that a "T" has been deleted and that the aligned region has been moved.*

7.1.10 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:

Right-click the name "Consensus" | Open Sequence | Save (⬅)

This will make it possible to use this sequence for further analyses in the *CLC Main 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 original contig, you will always be able to go back to it by choosing the Reference contig in the consensus sequence's history (see figure 7.13).
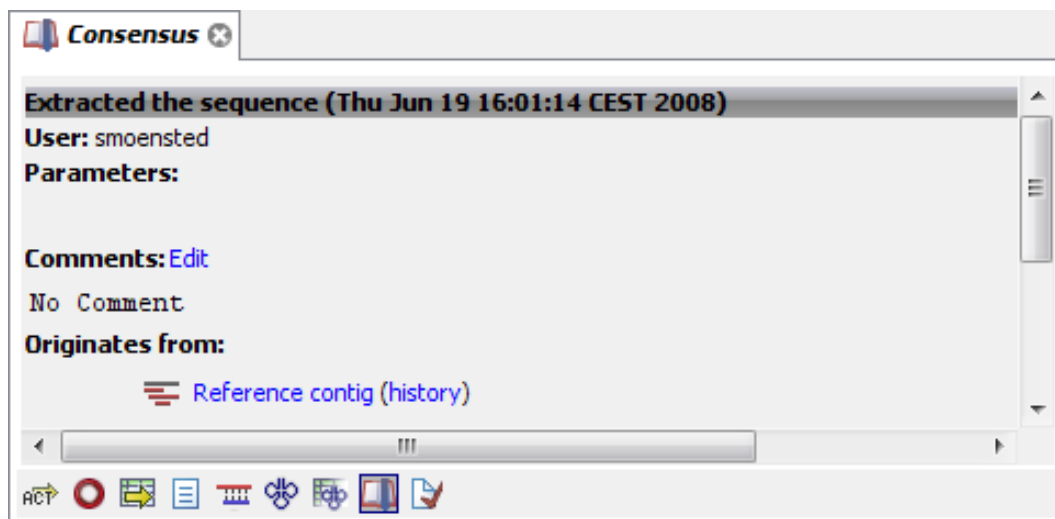
Figure 7.13: *The history of the consensus sequence, which has been extracted from the contig. Clicking the blue text "Reference contig" will find and highlight the name of the saved contig in the Navigation Area. Clicking the blue text "history" to the right will open the history view of the earlier contig. From there, you can choose other views, such as the Read mapping view, of the contig.*

7.2 Tutorial: In Silico Cloning Workflow

In this tutorial, the goal is to make a virtual PCR-amplification of a gene using primers with restriction sites at the 5' ends, and to insert the gene in a multiple cloning site of an expression vector. We start off with a set of primers, a DNA template sequence and an expression vector loaded into the Workbench.

This tutorial will guide you through the following steps:

1. Adding restriction sites to the primers

2. Simulating the effect of PCR by creating the fragment to use for cloning.

3. Specifying restriction sites to use for cloning, and inserting the fragment into the vector

7.2.1   Locating the data to use

Open the `Example data` folder in the **Navigation Area**. Open the `Cloning` folder, and inside this folder, open the `Primers` folder.

If you do not have the example data, please go to the **Help** menu to import it.

The data to use in these folders are shown in figure 7.14.
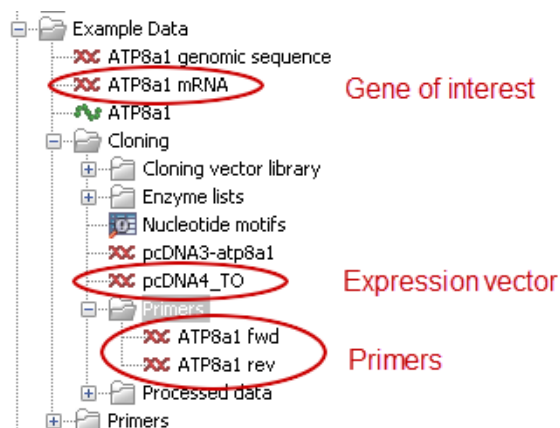
Figure 7.14: *The data to use in this tutorial.*

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 Main Workbench* (to learn more about this, please refer to the Primer design tutorial).

7.2.2   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 7.15.

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. If you do not see the whole sequence turn purple, please make sure you have the Selection Tool chosen, and not one of the other tools available from the top right side of the Workbench (e.g. Pan, Zooming tools, etc.)
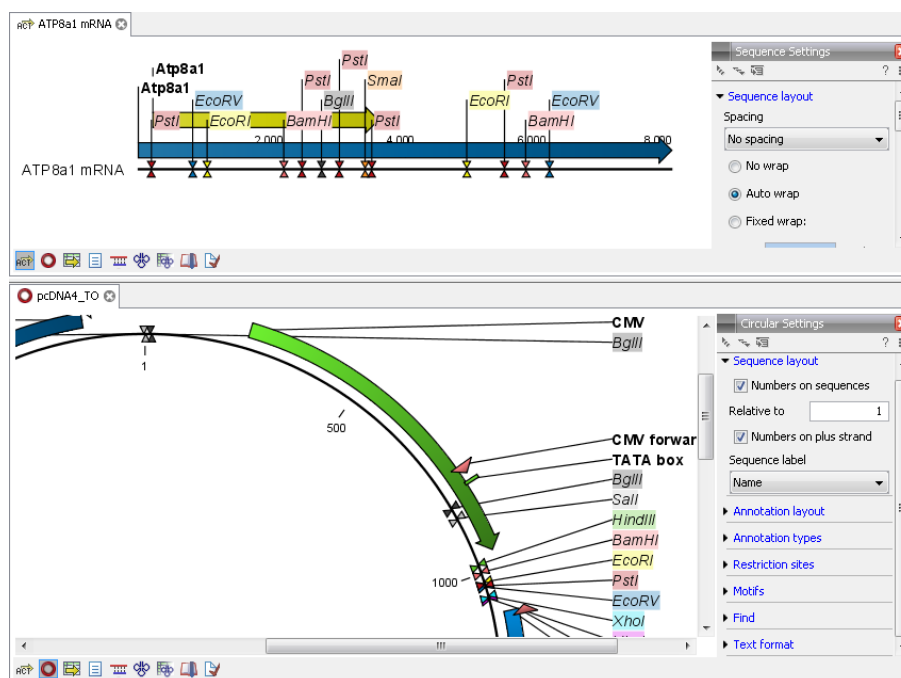
Figure 7.15: *Check cut sites.*

Once the sequence is selected, right-click and choose to **Insert Restriction Site Before Selection** as shown in figure 7.16.
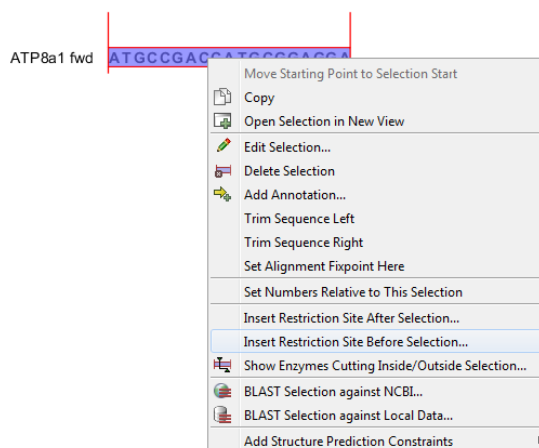
Figure 7.16: *Adding restriction sites to a primer.*

In the **Filter** box enter *HindIII* and click on 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 28.13.

Click **OK** and the sequence will be inserted at the 5' end of the primer as shown in figure 7.18.

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 figure28.13 when inserting the *HindIII* site.

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 (**Insert Restriction Site before Selection**).
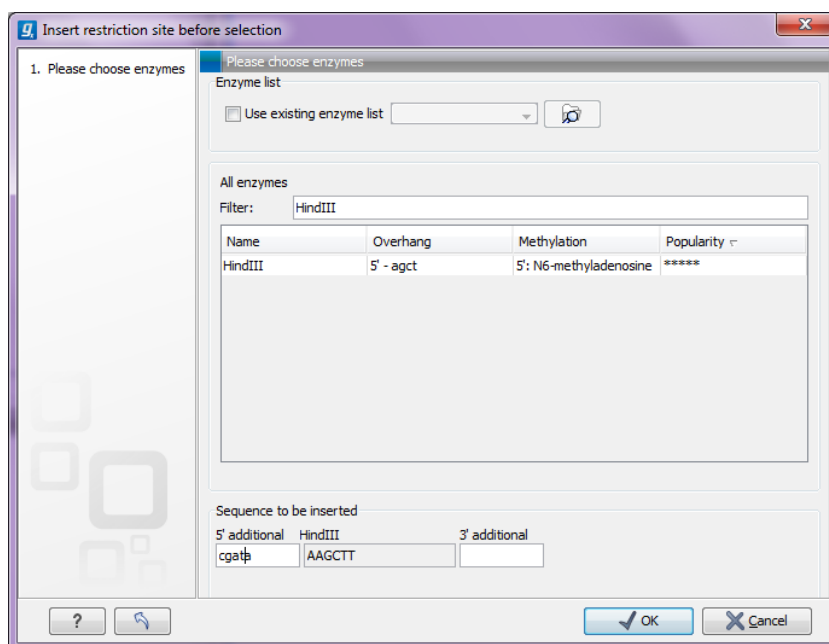
Figure 7.17: *Adding restriction sites to a primer.*

Figure 7.18: *Adding restriction sites to a primer.*

Save (⬅) the two primers and close the views and you are ready for next step.

7.2.3   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.

In the *CLC Main Workbench*:
> **Toolbox | Primers and Probes ()| Find Binding Sites and Create Fragments ()**

In the CLC Genomics Workbench:
> **Toolbox | Molecular Biology Tools () | 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 7.19.

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 7.20.

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.
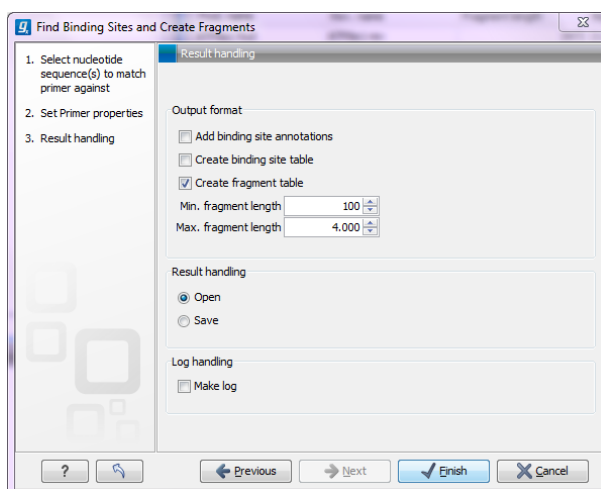
Figure 7.19: *Creating the fragment table including fragments up to 4000 bp.*
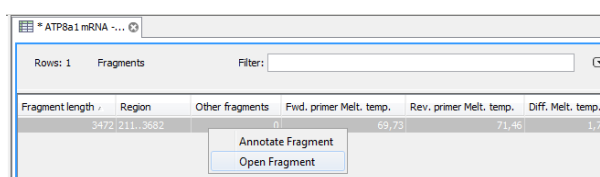
Figure 7.20: *Opening the fragment as a sequence.*

7.2.4   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 also select the `pcDNA4_TO` cloning vector also located in the `Cloning` folder and click **Next**. In this dialog,

The cloning editor will now create a sequence list of the selected fragment and vector sequence as shown in figure 28.2. Select the vector sequence and specify it as vector by clicking the "Change to Current" button at the top of the editor (indicated with a red circle in figure 28.2)

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 7.22).

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 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 7.23.

When this is done, the **Clone** (🔴) button at the lower right corner of the view is active because
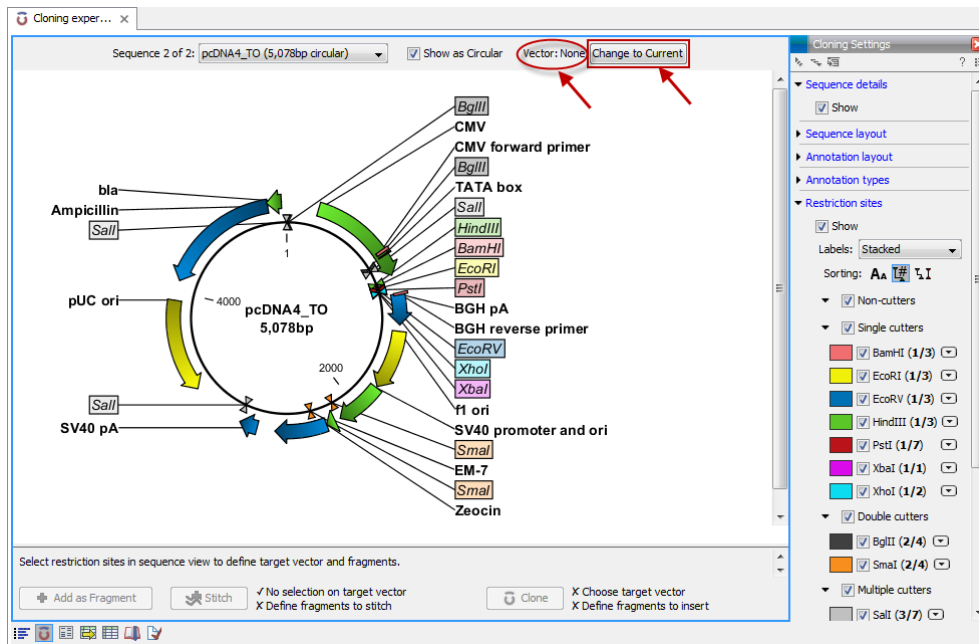
Figure 7.21: *Cloning editor.*
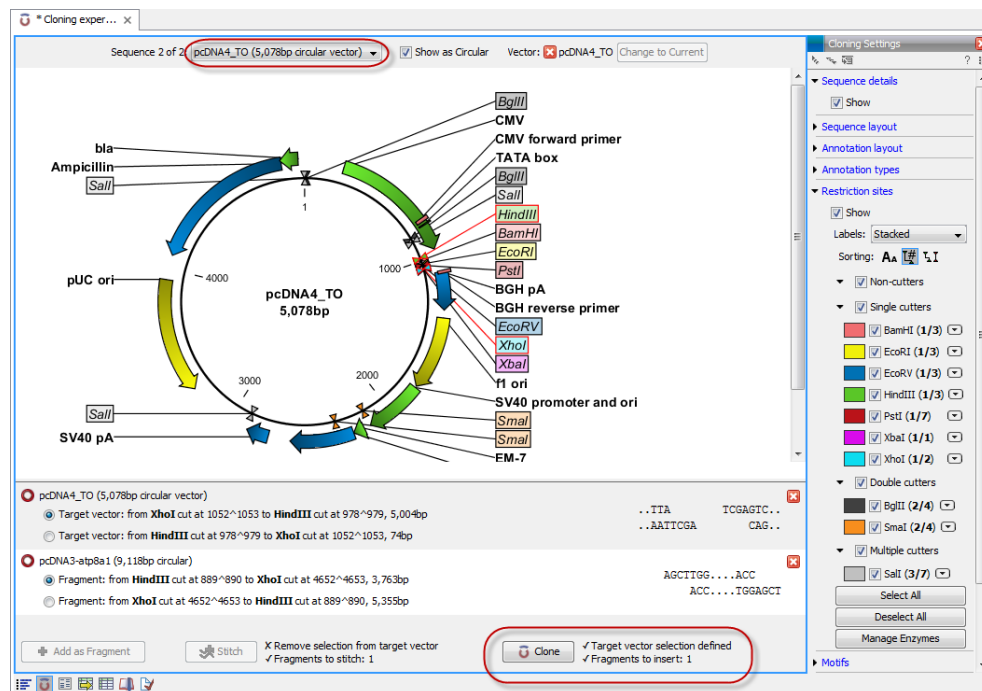
Figure 7.22: *Press and hold the Ctrl key while you click first the* HindIII *site and next the XhoI site in the cloning vector.*

there is now a valid selection of both fragment and target vector. Click the **Clone** ( ) button and you will see the dialog shown in figure 7.24.

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.

Figure 7.23: *Change the view to display the fragment sequence. Press and hold the Ctrl key while you click first the* HindIII *site and next the* XhoI *site.*

Figure 7.24: *Showing the insertion point of the vector*

When saving your work, there are two options:

- Save the Cloning Experiment. This is saved as a sequence list, including the specified cut sites. This is useful if you need to perform the same process again or double-check details.

- Save the construct shown in the circular view. This 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, 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 7.25.
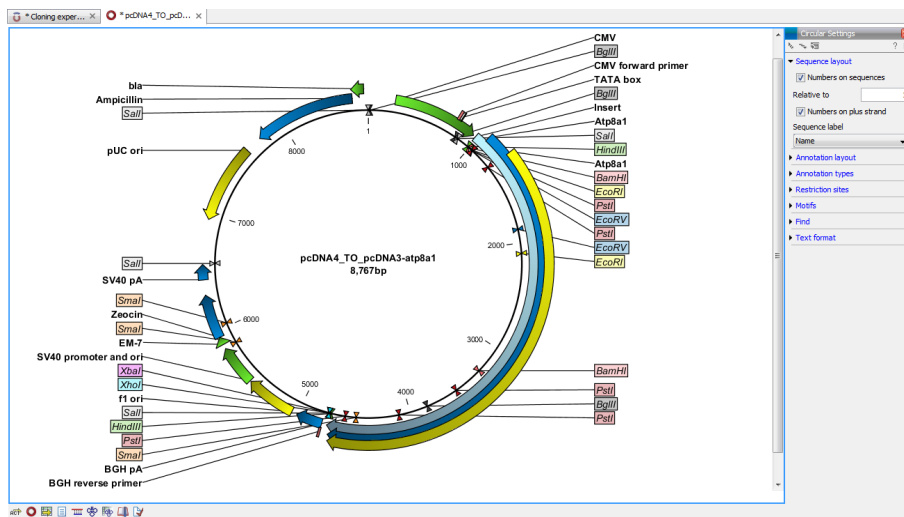
Figure 7.25: *The Atp8a1 gene inserted after the CMV promoter*

Chapter 8

Tutorial: Gateway Cloning

This tutorial will show you how to use the Workbench tool for in silico Gateway®Cloning. With this you can perform both standard (1 fragment) and mult-site Gateway Cloning. Here we focus on the 1-fragment cloning approach, but we provide tips to help you move onto multisite Gateway Cloning easily.

For information on the Gateway Cloning system please visit:

`http://www.invitrogen.com/site/us/en/home/Products-and-Services/Applications/`
`Cloning/Gateway-Cloning.html`.

After you have the Gateway Cloning vectors imported into your Workbench (section 8.1), you work through the three Gateway Cloning tools, which imitate the steps of the Gateway Cloning lab procedure. You will:

- Add attB sites to the fragment of interest, as would be done in the lab using PCR. (Section 8.2)

- Construct an entry vector by an attB/attP recombination event (BP reaction) in which the attB-flanked fragment is exchanged for the attP-flanked sequence of the chosen donor vector. (Section 8.3)

- Construct the final expression vector by an attL/attR recombination event (LR reaction) in which the target fragment of the entry vector is exchanged for the attR-flanked sequence of a chosen destination vector. (Section 8.4)

8.1 Importing Gateway Cloning vectors

Note! To be able to carry out the first steps of the tutorial (importing the .ma file) you first have to make sure that the **Vector NTI import** plugin is installed in your Workbench. For a description of how to install a plugin, please see the "Plugins" section in the CLC Genomics Workbench manual, which can be found here: `http://www.clcsupport.com/clcgenomicsworkbench/`
`current/index.php?manual=Installing_plug_ins.html`.

1. To perform the in silico Gateway Cloning we first need to import donor and destination vectors. A file of these can be downloaded from the Invitrogen web site:

> `http://tools.invitrogen.com/downloads/Gateway%20vectors.ma4`

2. Once downloaded, create a folder in your Workbench to import this file into.

3. Select this new folder by clicking on it.

4. Now, import the vector sequences into the Workbench using the **standard import tool. (**⬇**)**

The vector sequences should now be imported into your new folder.

For further information about the vectors for MultiSite Gateway Cloning, please refer to Invitrogen's web site that can be found here `http://www.invitrogen.com/`.

If you choose which type of MultiSite Cloning you wish to perform (number of fragments), then the relevant vectors can be listed. Vector information can then be downloaded from the linked areas on the site.

8.2 Adding attB sites

This step mimics the lab PCR procedure of adding attB sites to the target fragment making it eligible for recombination into the donor vector. The output can also include a set of primers, although we do not do that in this tutorial.

Our target fragment is the coding sequence (CDS) of ATP8a1, the mRNA of which is found in the standard example data you can download into the Workbench. If you do not already have the example data go to the **Help** menu in the Workbench, and choose to **Import Example Data**.

Now:

1. Extract the CDS sequence from the ATP8a1 mRNA file. To do this,

   - Double click on the sequence object called **ATP8a1 mRNA**. This opens it in the viewing area of the Workbench.
   - Right-click the CDS annotation, which by default will be coloured in yellow, and chose **Open annotation in New View**
   - Save the new data object that has just opened in the new view. There are different ways to do this. Two options are:
     (a) Right click on the tab at the top of the new view, and choose Save As... from the menu that appears, or
     (b) Move the cursor over the tab at the top of the new view, press the left mouse button. Keeping the mouse button down, drag the tab over into a folder in the **Navigation Area** of the Workbench. The data will then be saved into that folder.

2. Add the attB sites. To do this,

   - Start the **Gateway Cloning** tool:
     Toolbox | Cloning and Restriction Sites (🖼**)| Gateway Cloning | Add attB Sites (**〰**)**

- In the wizard, select the ATP8a1 CDS fragment you just saved. When the wizard starts up, that fragment should be in the list of **Selected Elements**. If it is not, please select it in the wizard, and click on the right hand arrow to add it to the **Selected Elements** list.

- Click on the button marked **Next**.

- It is in this step of the wizard you choose the attB sites to be added to your fragment. This is shown in figure 8.1. The sites to choose depend on your choice of donor vector. We will be going with the default option of attB1 and attB2. This ensures compatibility with our downstream choice of donor vector, which will be **pDONR221**.

- Click on the button marked **Next**.
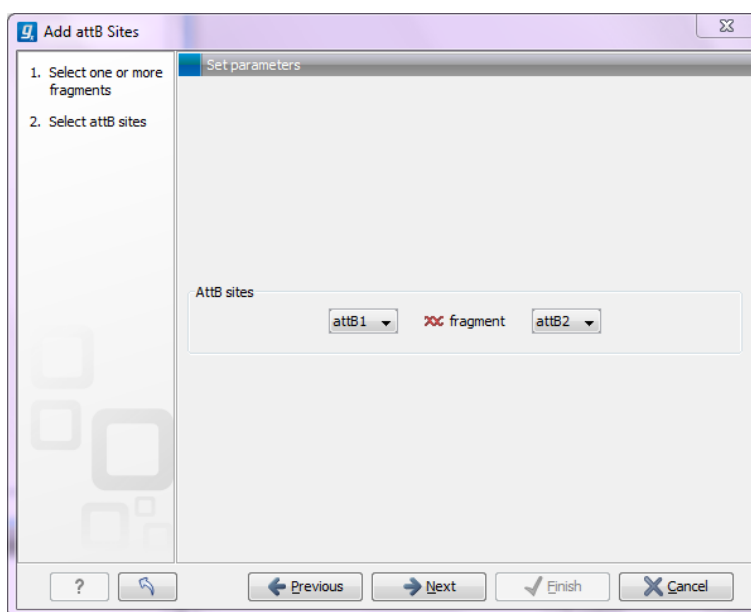
Figure 8.1: *Adding attB sites.*

If you have selected more than one fragment these will all be given the same combination of attB sites. If you wish to add different combination of sites you will have to run the tool once for each combination.

3. Next you will be given the option to insert additional elements downstream and upstream of the first and second attB site, respectively. That is, to insert elements immediately 5 prime and 3 prime of the fragment.

   We will add a Shine-Dalgarno site 5 prime of ATP8a1. Click in the field **Forward primer additions** and press on the keys Shift+F1 (for Mac: Shift + Fn + F1) to bring up a list of elements you can choose from. Use either the mouse or the up and down arrow keys and enter to select the Shine-Dalgarno site. See figure 8.2.

4. Insert 8 random nucleotides after the Shine-Dalgarno sequence. This is done by simply typing in the bases. See figure 8.3. Doing this ensures that the spacing between the Shine-Dalgarno sequence and the initiation codon is correct.

   You will find a preview of the final PCR product below. Any preset additional elements appear in green whereas the bases typed in manually are displayed in blue.

5. Click on the button marked **Next**.
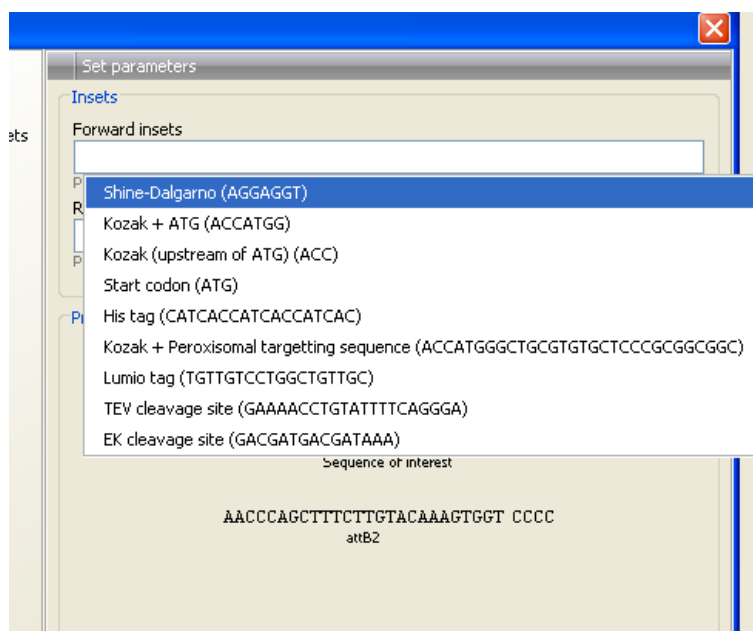
Figure 8.2: *Choose Shine-Dalgarno.*
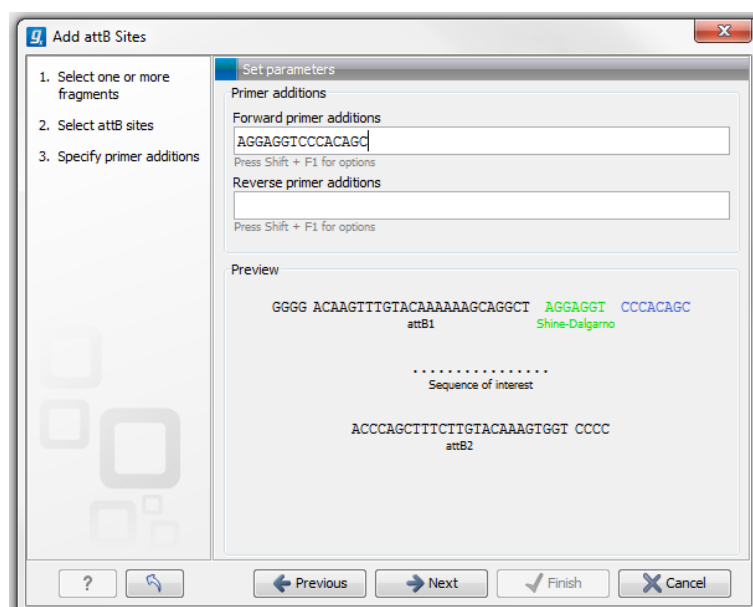
Figure 8.3: *Add bases to ensure correct spacing between the Shine-Dalgarno sequence and the initiation codon.*

6. In this step, you can choose the length of the template-specific part of your primers. See figure 8.4.

   This template-specific fragment is joined with the attB site and any additional elements you choose to make up the primer. Here, we will just keep the defaults.

7. Click on the button marked **Next**.

8. Choose to **Save** the results. Uncheck the **create primer list** option as we will not be needing this in this tutorial. See figure 8.5.

9. Click on the button marked **Next** and choose where to **Save** the output.
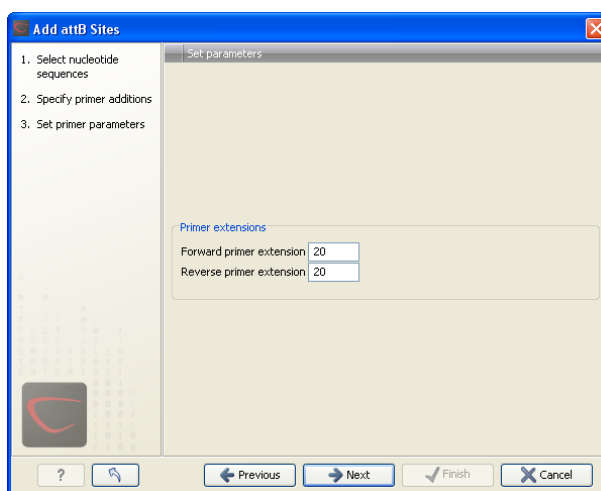
Figure 8.4: *Set the appropriate primer lengths.*
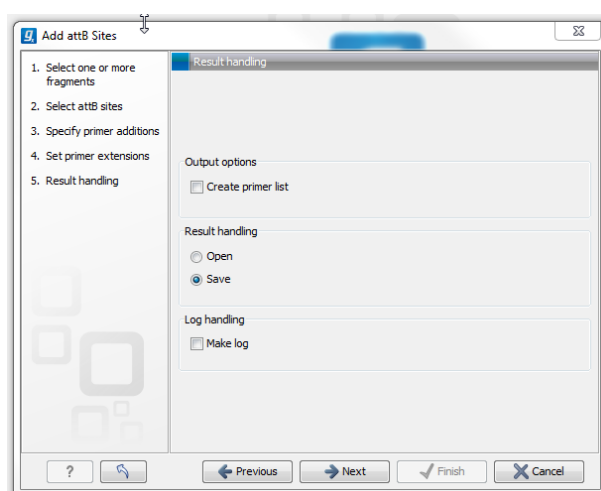
Figure 8.5: *Choose to save your results.*

10. Open up the result **ATP8a1 CDS (attB1 attB2)** to have a look. See figure 8.6.

11. In the side panel called **Sequence Settings**, find the section called **Annotation types**. Click on it to expand it.

12. Click on the box beside **RBS** to mark it, so that the Shine-Dalgarno ribosomal binding site annotation is displayed. Note also that you can see annotations for primer regions and the attB sites.

8.3   Creation of an entry vector

This step recombines the attB-flanked ATP8a1 into the donor vector, which is **pDONR221**, resembling the BP-reaction carried out in the lab.

1. Choose **Create Entry Clone** from under the Gateway Cloning menu:

   Toolbox | **Cloning and Restriction Sites (** **)**| **Gateway Cloning** | **Create Entry Clone (** **)**
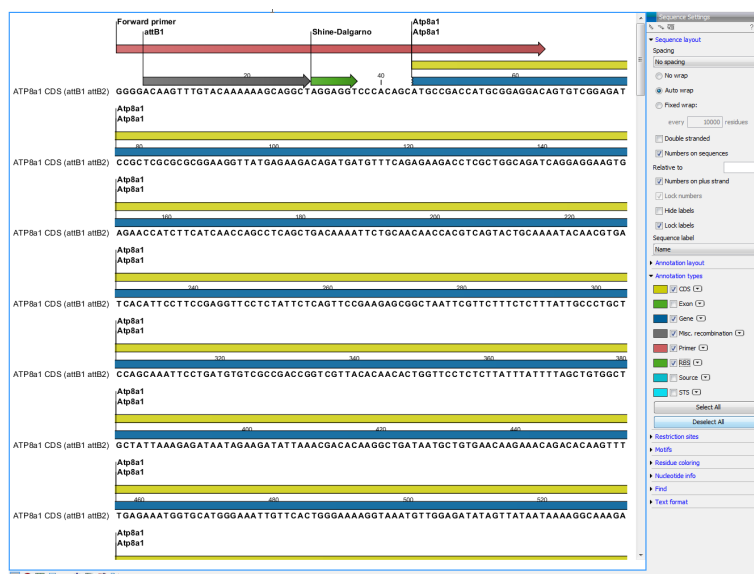
Figure 8.6: *A view of the results: ATP8a1 CDS (attB1 attB2).*

2. In the dialog select the result that you generated earlier: **ATP8a1 CDS (attB1 attB2)**. Then click on the button marked **Next**.

3. Click the **Browse button** (🔍) and find the Gateway folder of vectors, which was previously imported.

4. Select donor vector **pDONR221** as shown in figure 8.7 and click on the button labelled **OK**.
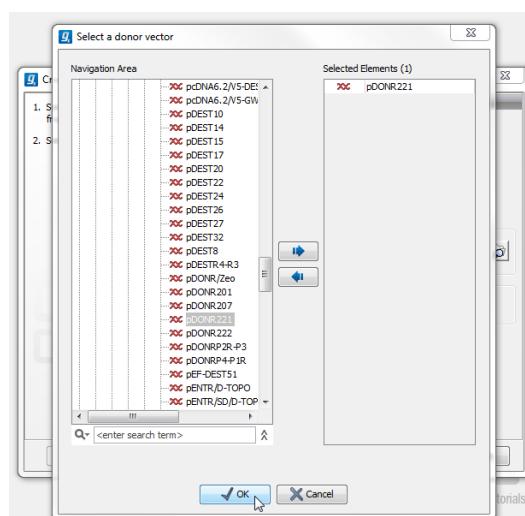
Figure 8.7: *Select the donor vector, which in this case is pDONR221.*

5. Click on the button labelled **Next**.

   Note: The Workbench only checks the chosen entry clone for valid attP sites. It does **not** check that these are compatible with the attB sites present in the fragment. You need to make sure that a valid entry vector is used for the attB-fragment of interest. If the right combination of attB and attP sites is not found, no entry clones will be produced.

6. Choose to **Save** the result and click on the button labelled **Next** to be able to specify where to save the results to.

7. Click on the button labelled **Finish**.

8. Open the output **pDONR221 (ATP8a1 CDS (attB1 attB2))** to have a look at it. To see a view like that shown in figure 8.8:

   - Go to circular view ( ).

   - Check annotation type **RBS** in the side panel to get a better view. The attB/attP recombination event has resulted in the fragment of interest being inserted into the vector. The fragment is flanked by attL sites.
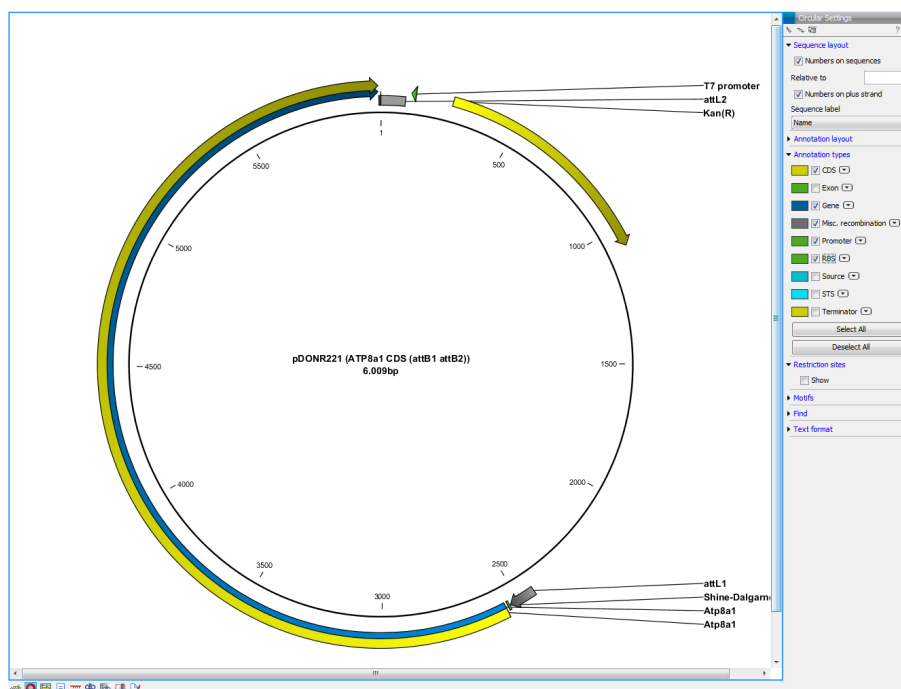
Figure 8.8: *View of the entry clone, consisting of the donor vector, pDONR221, with the fragment of interest inserted by attB/attP recombination.*

If several fragments were selected for input (using the batch option), the output would be one entry clone per fragment, all based on the same donor vector.

8.4 Creating an expression vector (LR)

This last step in a Gateway Cloning workflow mimics the lab LR recombination reaction. That is, the fragment of interest is recombined from the entry vector into a destination vector (the LR reaction) to create the final expression vector.

In this tutorial we are carrying out a standard, single fragment Gateway Cloning task. Thus we will choose only one entry clone as input during this step. For MultiSite Gateway Cloning, we would choose all the entry clones to be joined in the final expression vector. Note, that if you do this, you need to ensure that the entry clones contain a valid combination of attL sites. For more information on this please have a look at the MultiSite Gateway explanation on the Invitrogen webpage:

http://tools.invitrogen.com/downloads/gateway-multisite-seminar.html

The steps to create an expression clone are:

1. Choose **Create Expression Clone** from under the Gateway Cloning menu:

   Toolbox | Cloning and Restriction Sites ( )| Gateway Cloning | Create Expression Clone ( )

2. In the dialog box select the entry clone you created in the previous step and click on the button marked **Next**.

3. In the next window, click the **Browse** button and navigate to the Gateway vector folder. Select the destination vector: **pDEST14** and click on the button labelled **OK**.

4. Click on the button labelled **Next**.

   The Workbench only checks for valid attR sites in the destination vectors. You need to ensure that the attL and attR sites are compatible. If not, no expression clone will be created.

5. In the next window choose to open the result. If you would like to also have a look at the byproduct, check the corresponding box. Click on the button labelled **Finish**.

The output is the result of the recombination between the attL and attR sites. The fragment of interest is now inserted in the expression vector flanked by attB sites, see figure 8.9
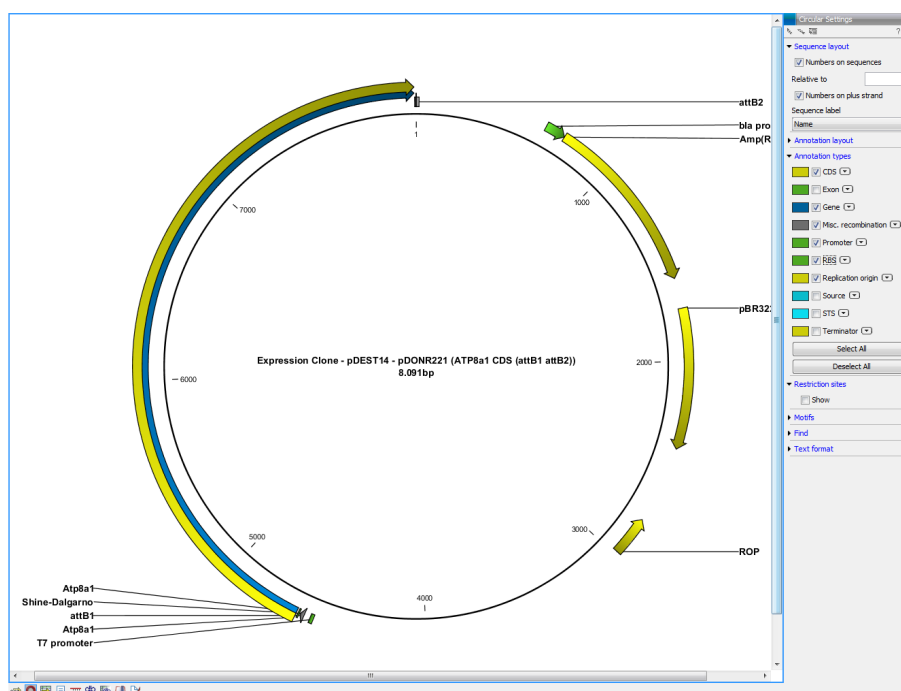
Figure 8.9: *View the finished expression vector.*

This concludes the Gateway Cloning tutorial. Below is a section outlining the general steps you could take in the case of MultiSite Gateway Cloning.

8.5   Short suggestion for a MultiSite Gateway Workflow using 2 fragments

The vectors mentioned in this section can be downloaded from Invitrogens web page:

http://tools.invitrogen.com/downloads/Gatewayvectors.ma4

1. Choose two fragments, which you would like to have in the final vector.

2. Create the following fragments:

   - Add attB1 and attB5r sites to sequence of interest 1 –> sequence of interest 1 (attB1 attB5r)

   - Add attB5 and attB2 sites to sequence of interest 2 –> sequence of interest 2 (attB5 attB2)

3. Create two entry clones:

   - sequence of interest 1 (attB1 attB5r) + pDONR221-P1P5r

   - sequence of interest 2 (attB5 attB2) + pDONR221-P5P2

4. Create the final expression vector by selecting:

   - the first entry clone: pDONR221-P1P5r (fragment of interest 1(attB1 attB5r)), and

   - the second entry clone: pDONR221-P5P2 (fragment of interest 2(attB5 attB2))

   - the destination vector: pDEST14

8.6   Tutorial: Primer Design

In this tutorial, you will see how to use the *CLC Main Workbench* to find 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 design primers that would allow us to generate a PCR product covering the insertion point of the gene. This would let us use PCR to check that the gene is 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.

8.6.1   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. This may be hidden behind restriction site annotations. Remember that you can always choose not to Show these by altering the settings in the right hand pane.
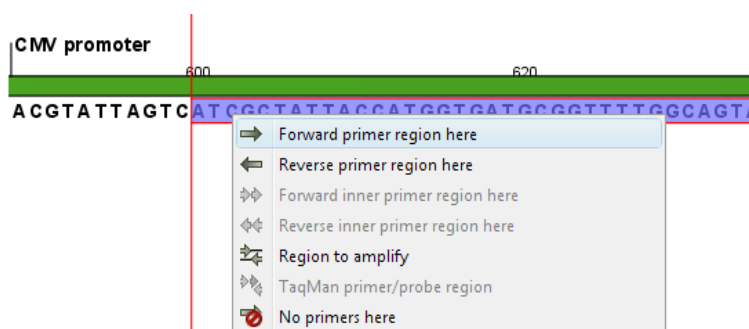
Figure 8.10: *Right-clicking a selection and choosing "Forward primer region here".*

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 8.10).

This will add an annotation to this region, and five rows of red and green dots are seen below as shown in figure 8.11:

Figure 8.11: *Five lines of dots representing primer suggestions. There is a line for each primer length - 18bp through to 22 bp.*

8.6.2   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 8.12):

Figure 8.12: *The first dot on line one represents the starting point of a primer that will anneal to the highlighted region.*

Position the mouse cursor over a dot. A box will appear, providing data about this primer. Clicking the dot will select the region where that primer would anneal. (See figure 8.13):

Note that some of the dots are colored red. This indicates that the primer represented by this
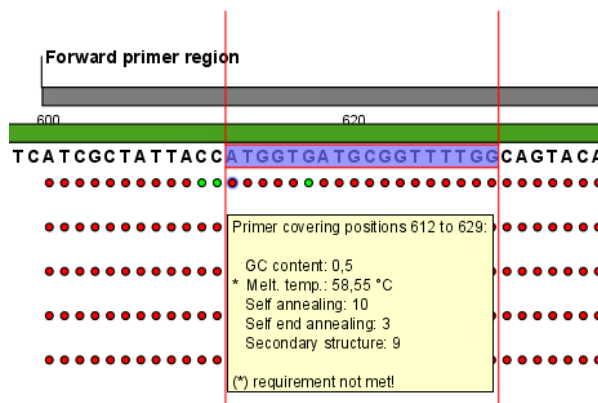
Figure 8.13: *Clicking the dot will select the corresponding primer region. Hovering the cursor over the dot will bring up an information box containing details about that primer.*

dot does not meet the requirements set in the **Primer parameters** (see figure 8.14):
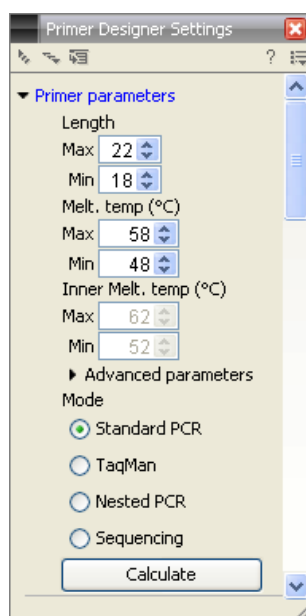
Figure 8.14: *The Primer parameters.*

The default maximum melting temperature is 58. This is the reason why the primer in figure 8.13 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 8.13 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 see why a specific primer (represented by a dot) fails to meet the requirements.

By adjusting the **Primer parameters** you can define primersto meet your specific needs. Since the dots are dynamically updated, you can immediately see how a change in the primer parameters affects the number of red and green dots.

8.6.3 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 8.15:
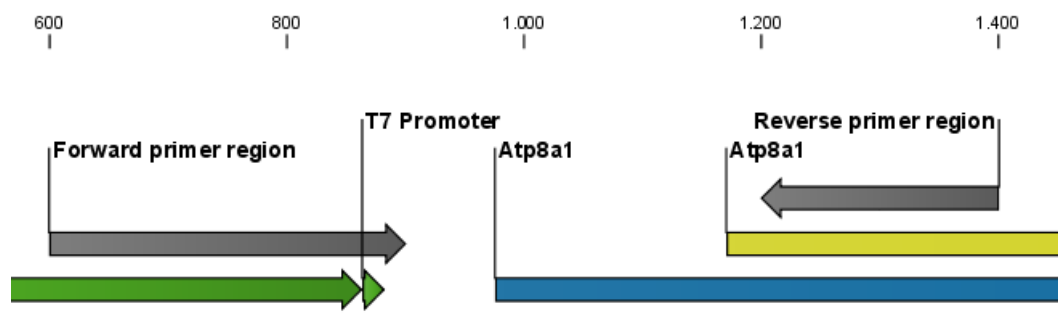
Figure 8.15: *A forward and a reverse primer region.*

Now, you can let *CLC Main Workbench* calculate all the possible primer pairs based on the **Primer parameters** that you have defined:

Click the Calculate button (right hand pane) | 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 8.16):

Figure 8.16: *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 8.17):

This tutorial has shown some of the many options of the primer design functionalities of *CLC Main Workbench*. You can read much more using the program's **Help** function (?) or in the *CLC Main Workbench* user manual, linked to on this webpage: http://www.clcbio.com/download.

Figure 8.17: *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").*

8.7 Tutorial: Working with Annotations

Annotations are the basis of many of the analyses in the Workbench. Once an annotation has been added to a sequence, it stays there even when the sequence is transformed to be part of an alignment, a BLAST result or a sequence list.

Because annotations are so fundamental to many of the tasks performed in the Workbench, there are a number of ways to browse, search for, view and edit annotations. This tutorial takes you through a lot of different areas of the Workbench to show some of the places where you can work with annotations.

8.7.1 Browsing and viewing annotations in sequence views

Open sequence *ATP8a1 mRNA* from the Example data by double-clicking the sequence. Click **Annotation types** in the **Side Panel** and you will see a list of the types of annotations on this sequence (shown in figure 8.18).
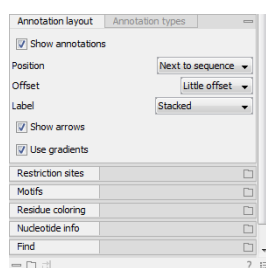
Figure 8.18: *Annotation types of the ATP8a1 mRNA sequence.*

Click the *Exon* annotation type. This will show exon annotations on the sequence. As this

sequence is very long, you cannot see the whole sequence, and the exons may therefore not be visible. In this situation, you can either Zoom out to find the annotations, or you can use the browse button ( ) next to the annotation type to jump directly to one of the annotations (see figure 8.19).
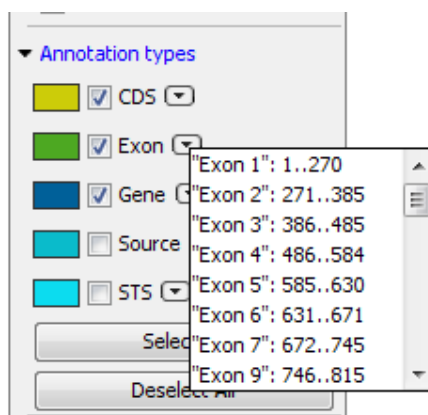
Figure 8.19: *Click in the list to go directly to this annotation.*

Click *Exon 2*, and the beginning of the annotation will be visible in the view (see figure 8.20).
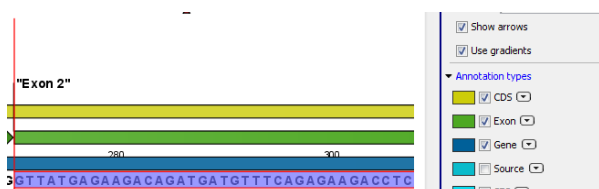
Figure 8.20: *The view jumps to the beginning of exon 2.*

Annotations are displayed as arrows, going from left to right on the positive strand, and going from right to left on the negative strand. Placing your mouse cursor on the arrow will show additional information about the annotation. Try to place the mouse cursor on the blue gene annotation, and you can see more information about it (see figure 8.21).
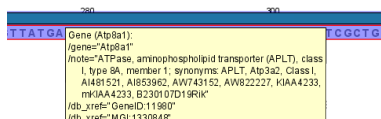
Figure 8.21: *Information about the annotation.*

This way of displaying and accessing annotations is similar for both circular views, alignments, and all other views displaying sequence residues.

8.7.2 Adding and editing annotations

You can add your own annotations to a sequence. A new annotation is most easily added if you first select a region on the sequence. In the upper view, select a region from residue 10 to 26 (see the status bar in the lower right corner of the Workbench). Then:

> **right-click the selection | Add Annotation ( )**

This will display a dialog where you can enter more information about the annotation. Enter *Test* as the name of the annotation (see figure 8.22).
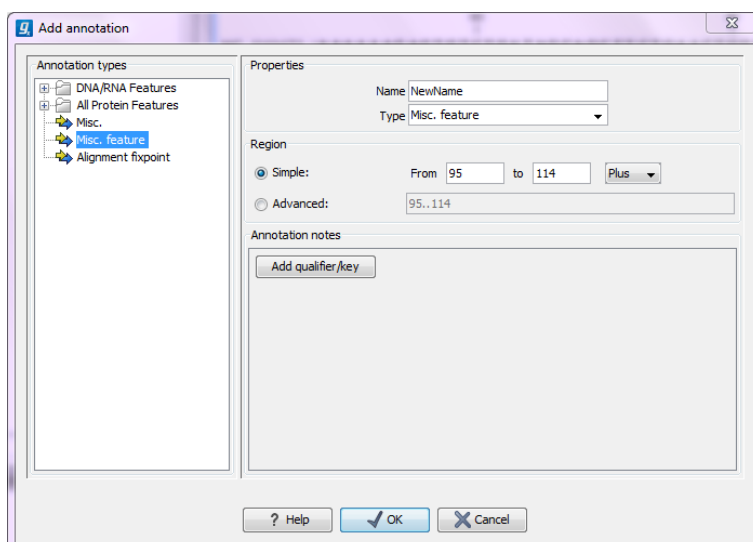
Figure 8.22: *The Add Annotation dialog.*

Click **OK** and the annotation is added to the sequence (see figure 8.23).
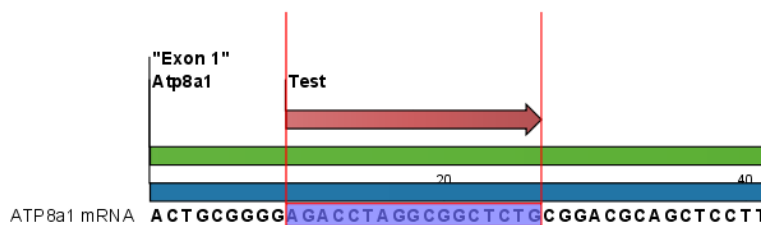
Figure 8.23: *The annotation has been added.*

Now click the **Show Annotation Table** (🔳) icon at the bottom of the view. This will display a list of all the annotations on the sequence.

Enter *Test* into the **Filter** in the annotation table, and your newly added annotation will appear.

You can now double-click any of the **Name**, **Type** or **Region** fields to edit the annotation. Double-click the **Type** which is now "Misc. feature" and enter *My type* and press **Enter**.

The table should now look as shown in figure 8.24.
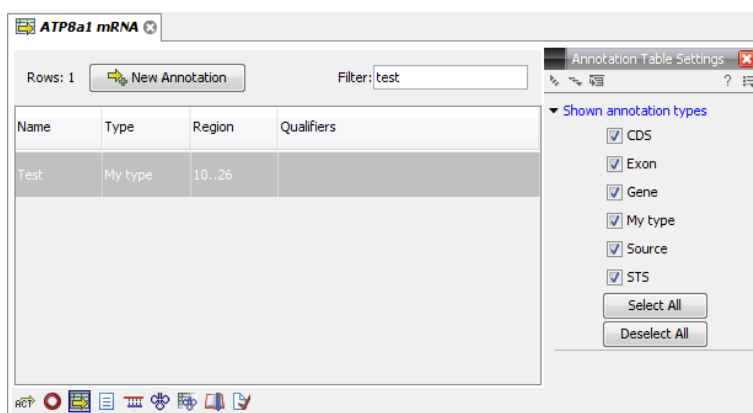
Figure 8.24: *The type of the annotation is now My type.*

In this way, you can quickly edit annotations. If you double-click the **Qualifiers** cell, a dialog similar to the one in figure 8.22 will be shown.

8.7.3   Copying annotations

One of the more advanced features of the Workbench is the ability to copy and paste annotations. Annotations can be copied from one sequence to another or you can create a duplicate of the annotation by pasting it back on the same sequence. Because it is very easy to edit the region of the annotation, this can be a quick way of adding a number of similar annotations.

> **Select the *Test* annotation in the table | Copy (⬚) | Paste (⬚)**

Double-click the region of the new annotation and type *30..46* and press **Enter**. You now have two annotations as shown in figure 8.25.
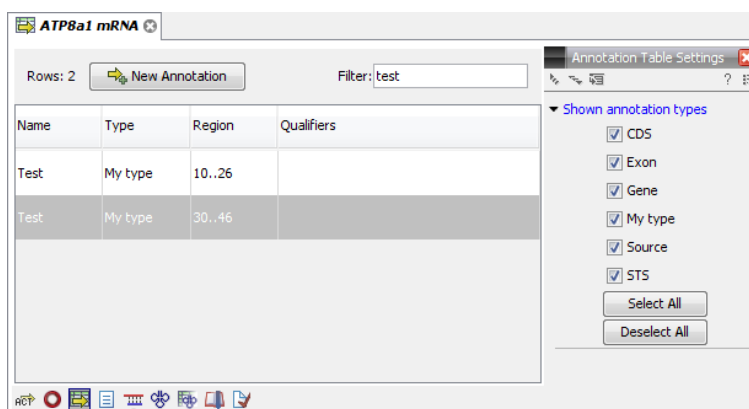
Figure 8.25: *Two copies of the Test annotation.*

8.8   Tutorial: BLAST

BLAST is an invaluable tool in bioinformatics. It has become central to identification of homologues and similar sequences, and can also be used for many other different purposes. This tutorial takes you through the steps of running a blast search in CLC Workbenches. If you plan to use BLAST for your research, we highly recommend that you read further about it. Understanding how BLAST works is key to setting up meaningful and efficient searches.

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, different nucleotide or protein BLAST query options can be performed against NCBI, in data subsets hereof or in local databases:

> **Toolbox | BLAST (⬚)**

- BLAST

- BLAST at NCBI

- Download BLAST Databases

- Create BLAST Database

- Manage BLAST Database

To obtain more information about this molecule you wish to query the peptides held in the Swiss-Prot* database to find homologous proteins in humans *Homo sapiens*, using the **Basic Local Alignment Search Tool** (BLAST) algorithm.

Please note: This tutorial involves running BLAST remotely using databases housed at the NCBI. Your computer must be connected to the internet to complete this tutorial.

8.8.1 Performing the BLAST search in own sequence data or own BLAST database

Select your query sequence(s), go to the Toolbox and select the BLAST search:

       select e.g. ATP8a1 genomic sequence | Toolbox | BLAST (📚**)**

As the tool works with either nucleotide or protein query sequences, it is important to select the relevant BLAST search program:

BLAST programs for DNA query sequences:

- **blastn: DNA sequence against a DNA database.** Used to look for DNA sequences with homologous regions to your nucleotide query sequence.

- **tblastn: Translated DNA sequence against a Translated DNA database.** Automatic translations of your DNA query sequence and the DNA database, in six frames. The resulting peptide query sequences are used to search the resulting peptide database. Note that this type of search is computationally intensive.

- **blastx: Automatic translation of your DNA query sequence in six frames;** these translated sequences are then used to search a protein database.

BLAST programs for protein query sequences:

- **tblastn: Protein sequence against Translated DNA database.** Peptide query sequences are searched against an automatically translated, in six frames, DNA database.

- **blastp: Protein sequence against Protein database.** Used to look for peptide sequences with homologous regions to your peptide query sequence.

The sequence data to be searched within is called the Target. This can either be locally stored sequences or a local database BLAST database (see subsequent sections on how to Create and Manage Local Blast Databases, respectively).

Note: Selected subsets of sequence will be used as a temporary database, and not be accessible afterwards.

Click **Next**.

A number of optional parameters can be set prior to performing the BLAST search (figure 8.27) e.g. number of threads, filter, size of expected E-value, word size, matrix, gap cost, and maximum number of hits to be shown. In the current and earlier versions of the Workbench, the number of hits being reported is as specified, however, all high-scoring segment pairs (HSPs) for those hits are provided in the results. Thus, in cases where some hits have more than one HSP, the hit table will have more rows than the number of hits requested.
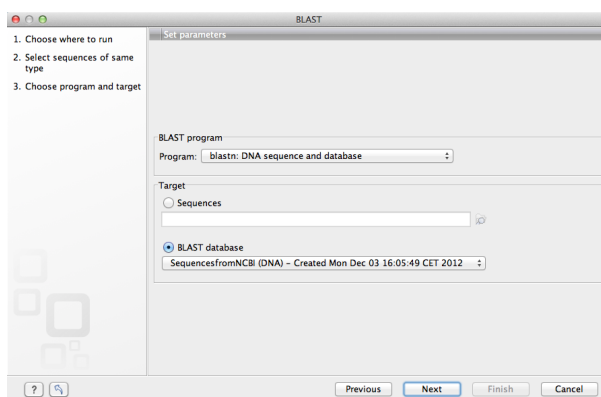
Figure 8.26: *The outcome of the BLAST query search is dependent on the BLAST program selected as well as the Target data to search within (locally stored sequences or BLAST database).*
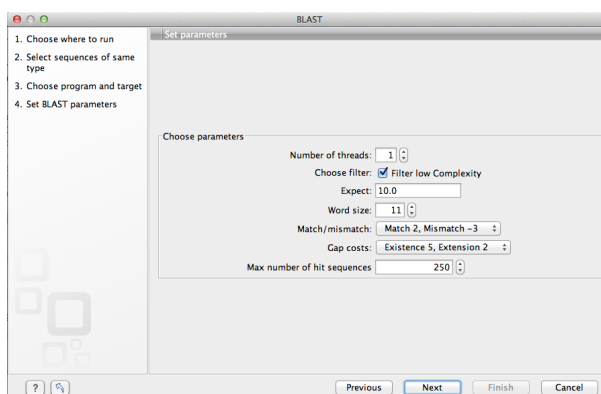
Figure 8.27: *A number of parameters can be modified to optimize the selected BLAST search.*

8.8.2 Create BLAST Database

BLAST databases can be created from DNA, RNA, and protein sequences located in the **Navigation Area**. Any given BLAST database can only include one molecule type. If you wish to use a pre-formatted BLAST database instead, see above section on Download BLAST Database.

To create a BLAST database, go to:

Toolbox | **BLAST** (📕) | **Create BLAST Database** (📗)
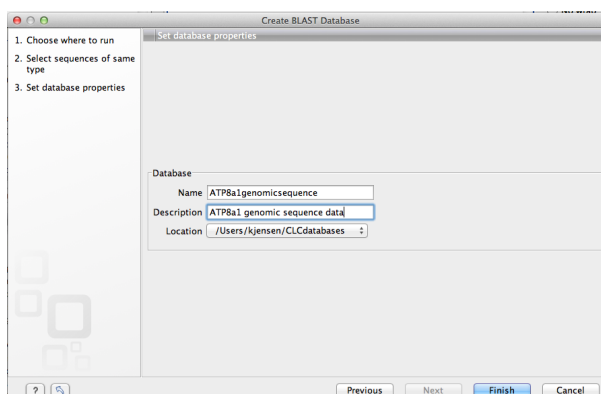
Figure 8.28: *When using Create BLAST Database, a database of the specified sequences files (and/or sequence lists) will be created.*

8.8.3 Download BLAST Database

The Download BLAST Database allows you to fetch copies of specific databases stored at NCBI FTP-site. Examples include 16S microbial sequences, environmental nucleotide or protein sequences, EST- or genomic data data from mouse, human, other organisms, ref sequences etc. (figure 8.29).

The download location can be specified, and downloaded databases can be managed subsequently using the Manage BLAST Databases (see following section).
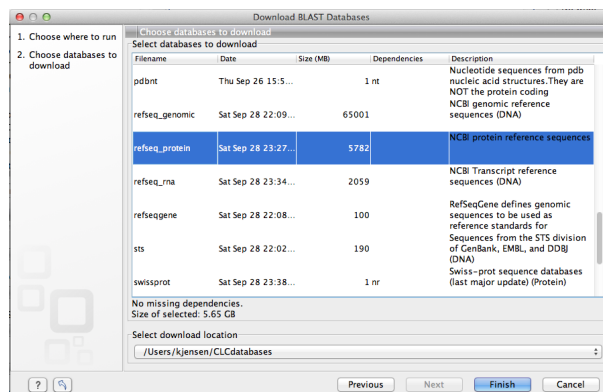
Figure 8.29: *Topic specific databases can be downloaded from NCBI's FTP server, and stored locally.*

8.8.4 BLAST Database Manager

By default a BLAST database location will be added under your home area in a folder called CLCdatabases. This folder is scanned recursively, through all subfolders, to look for valid databases. All other folder locations are scanned only at the top level. Once downloaded, it is easy to get an overview of the local databases as well as to modify them (add- and refresh location as well as remove database, see figure 8.30).
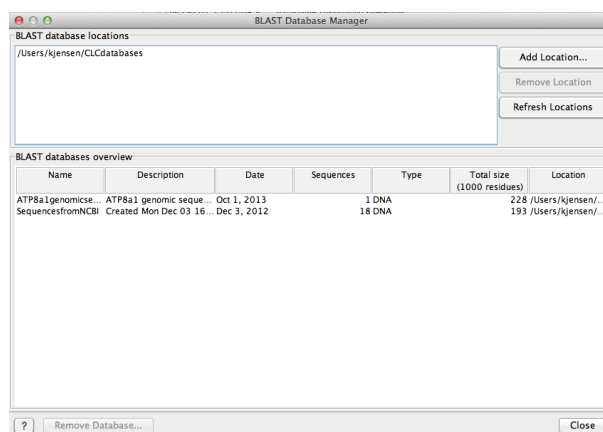
Figure 8.30: *The BLAST Database Manager provides an overview of available BLAST databases, their description, size and location, as well as the option to change location.*

8.8.5 Performing the BLAST search

Start out by:

select protein ATP8a1 | Toolbox | BLAST ( ) | BLAST at NCBI ( )

In **Step 1** you can choose which sequence to use as query sequence. AS you have already selected a sequence, the sequence is already displayed in the **Selected Elements** list.

Click **Next**.

In **Step 2** (figure 8.31), choose the default BLAST program: **blastp: Protein sequence and database** and select the **Swiss-Prot** database in the **Database** drop down menu.
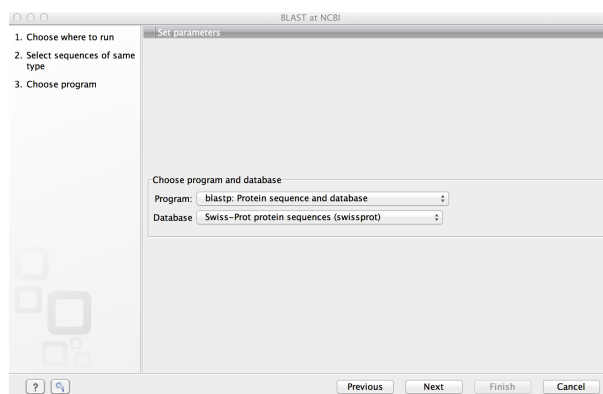
Figure 8.31: *Choosing BLAST program and database.*

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 8.32. Including this term will limit the query to proteins of human origin.
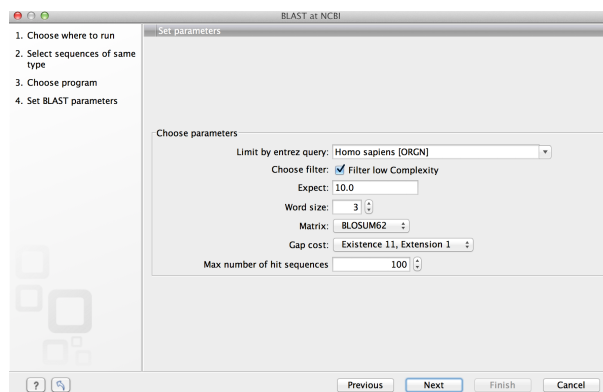
Figure 8.32: *The BLAST search is limited to homo sapiens[ORGN]. The remaining parameters are left as default.*

Choose to **Open** your results.

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 should be received and opened in a new view.

8.8.6   Inspecting the results

The output is shown in figure 8.33 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.
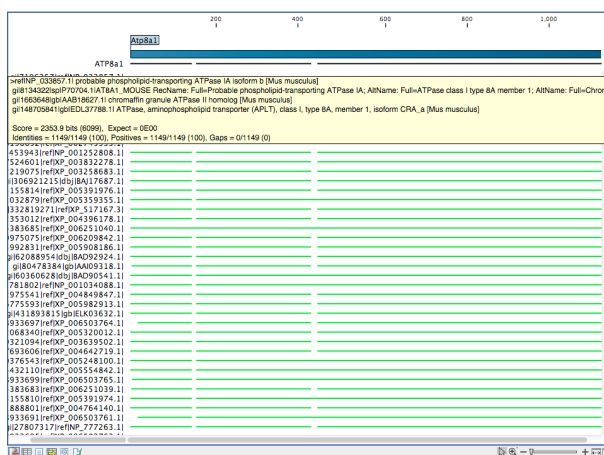
Figure 8.33: *Output of a BLAST search. By holding the mouse pointer over the lines you can get information about the sequence.*

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 that have been downloaded. This means that you can zoom in and see the actual alignment:

> **Zoom in using the Tool Bar zoom function ( ) | 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**

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 Main 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.

8.8.7   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 8.34.

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**.

Figure 8.34: *Output of a BLAST search shown in a table.*

8.9   Tutorial: Tips for Specialized BLAST Searches

Here, you will learn how to:

- Use BLAST to find the gene coding for a protein in a genomic sequence.

- Find primer binding sites on genomic sequences

- Identify remote protein homologues.

Following through these sections of the tutorial requires some experience using the Workbench, so if you get stuck at some point, we recommend going through the more basic tutorials first.

8.9.1   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 arrestin domain-containing protein 5 (ARRDC5) protein to a chromosome. We know in advance that the ARRDC5 is located somewhere on chromosome 19.

Data used in this example can be downloaded from GenBank:

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

Human chromosome 19 (NC_000019) consists of 59128983 nucleotides and the ARRDC5 (NP_001073992) protein has 342 amino acids.

BLAST configuration

Next, conduct a local BLAST search:

> **Toolbox | BLAST (** **) | BLAST (** **)**

Select the protein sequence as query sequence and click **Next**. Since you wish to BLAST a protein sequence against a nucleotide sequence, select **tblastn**, which will automatically translate the selected nucleotide sequence to database format.

Select the just downloaded chr19 sequence *NC_000019* as target. If you are used to BLAST, you will know that you usually have to create a BLAST database before BLASTing. However,the Workbench does this "on the fly" when one or more sequences have been selected.

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 8.35). This is done by pressing Ctrl (⌘ on Mac) while clicking the table view (▦) at the bottom of the view.

In the side panel under BLAST table settings, select the same settings as shown in figure 8.35 by checking the relevant parameters under "Show column".

Now, sort the BLAST table view by clicking twice on the column header "% Positive". Then, press and hold the Ctrl button (⌘ on Mac) and click the header "Query start". You have now sorted the table first on % Positive hits and then on the start position of the query sequence. In the table you can see three hits with 97 - 100% positive (=similar residues) at different locations on the chromosome sequence (see figure 8.35).
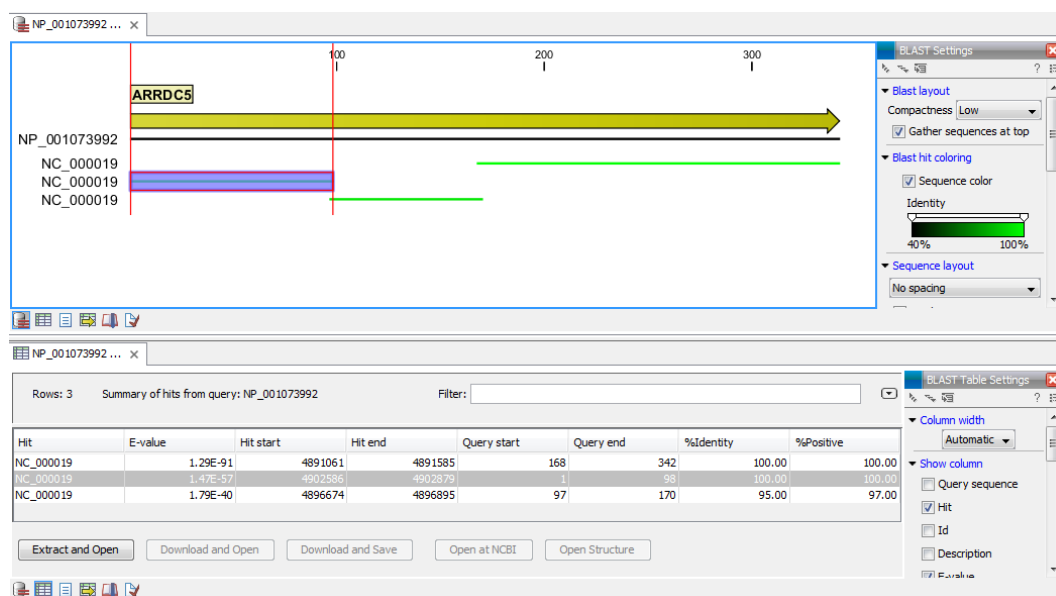
Figure 8.35: *Placement of translated nucleotide sequence hits on the Human ARRDC5 protein.*

Why did we find, on the protein level, three identical regions between our query protein sequence and nucleotide database?

The ARRDC5 gene is known to have three exons, which is in agreement with the three hits 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, -3 and -3 for the three hits, respectively.

Verify the result

Open NC_0000019 in a view, and go to Hit position (4,902,879) and zoom to see the blue gene annotation. You can now see the exon structure of the ARRDC5 gene showing the three exons on the reverse strand (see figure 8.36).
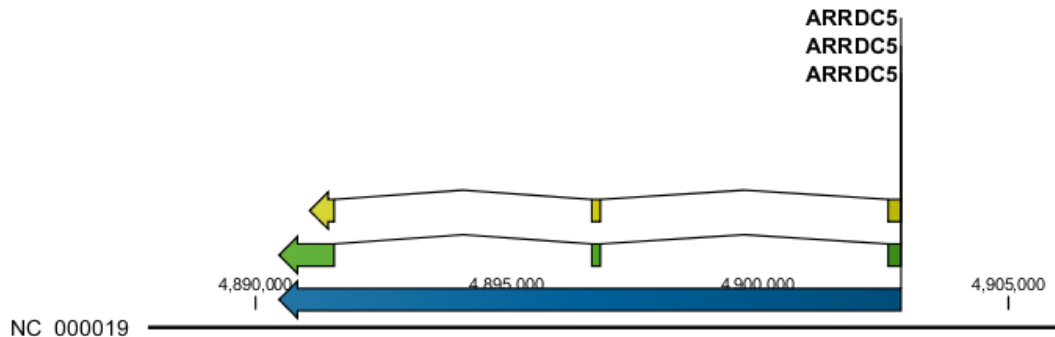
Figure 8.36: *ARRDC5 exon view.*

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, NP_001073992, as target sequence

Tip: In stead of using the protein sequence NP_001073992 as target sequence you can also use it as database. To do this, you first have to create a database:

> **Toolbox** | **BLAST ( )** | **Create BLAST database ( )**

Using the genomic sequence as query, the mapping of the protein sequence to the exons is visually very clear as shown in figure 8.37.

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.

8.9.2   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.
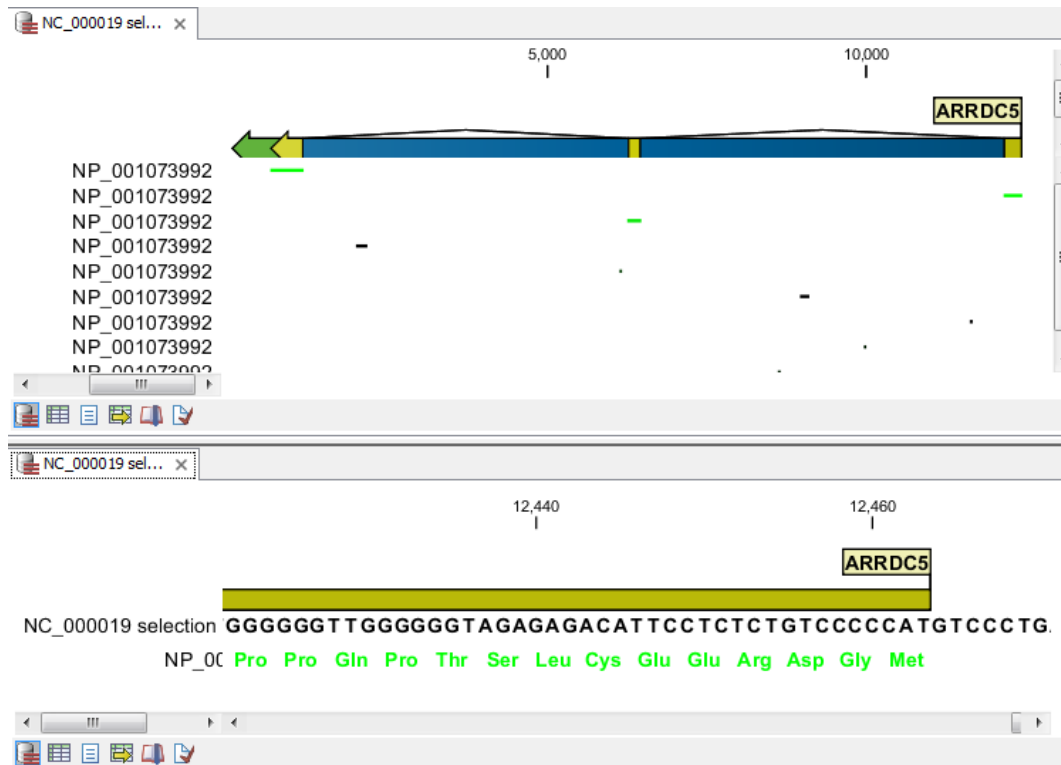
Figure 8.37: *Verification of the result: at the top a view of the whole BLAST result. At the bottom the same view zoomed in on exon 3 to show the amino acids.*

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 8.38.
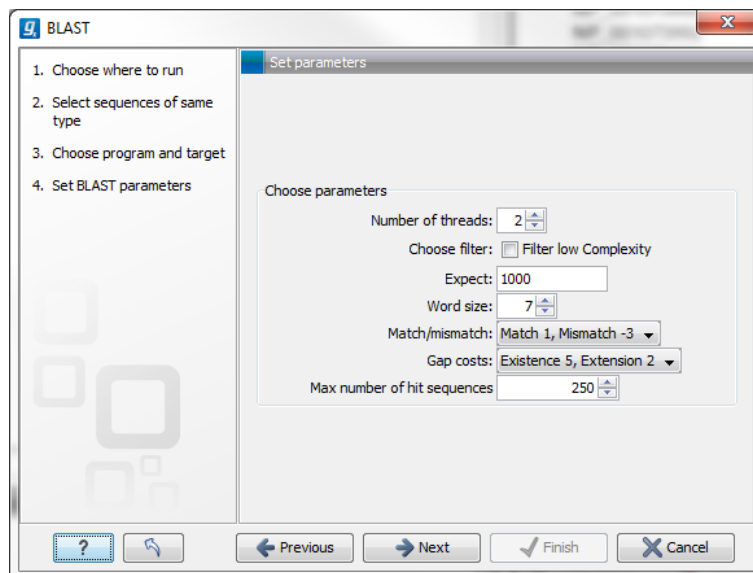
Figure 8.38: *Settings for searching for primer binding sites.*

8.9.3 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 8.39.
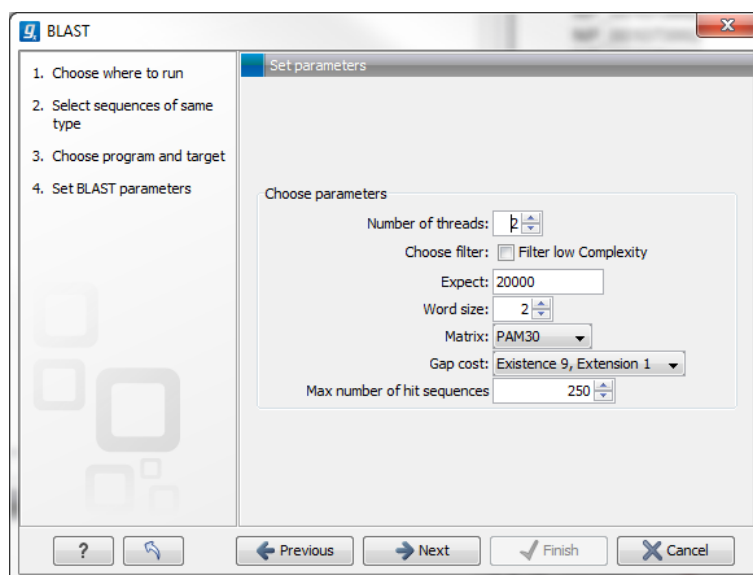
Figure 8.39: *Settings for searching for remote homologues.*

8.9.4 Further reading

A valuable source of information about BLAST can be found at http://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs&DOC_TYPE=ProgSelectionGuide.

Remember that BLAST is a heuristic method. This means that certain assumptions are made to allow searches to be done in a reasonable amount of time. Thus you cannot trust BLAST search results to be accurate. For very accurate results you should consider using other algorithms, such as Smith-Waterman. You can read "Bioinformatics explained: BLAST versus Smith-Waterman" here: http://www.clcbio.com/BE.

8.10 Tutorial: Proteolytic Cleavage

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 Analysis ( )| 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 8.40.
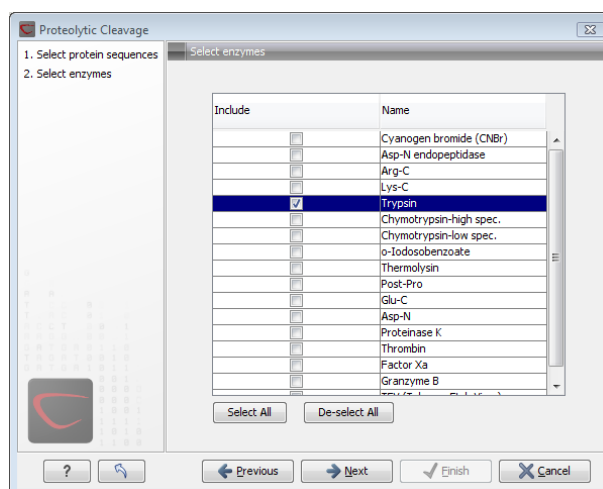
Figure 8.40: *Selecting trypsin as the cleaving enzyme.*

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 8.41:
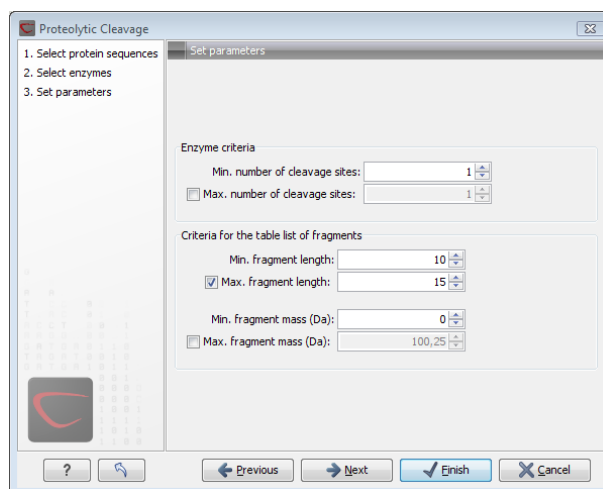
Figure 8.41: *Adjusting the output from the cleavage to include fragments which are between 10 and 15 amino acids long.*

Click **Finish** to make the analysis. The result of the analysis can be seen in figure 8.42

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
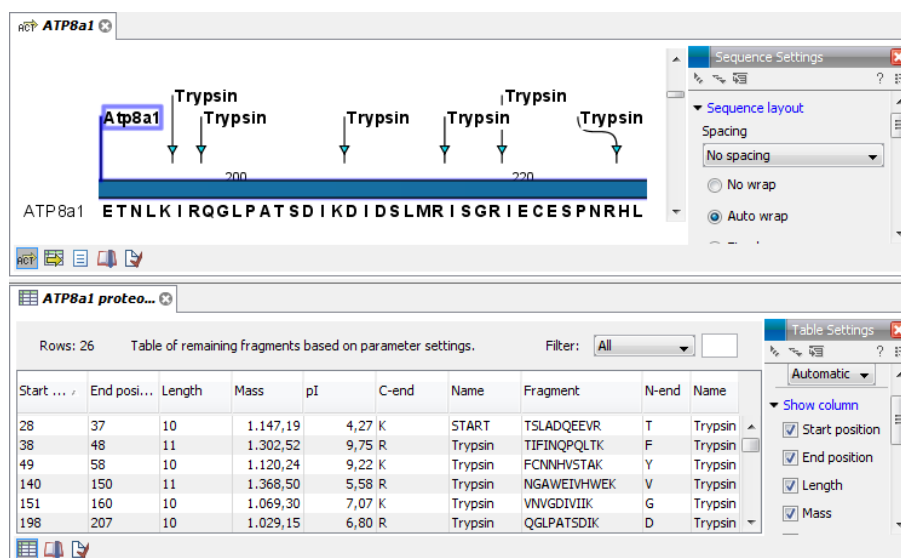
Figure 8.42: *The output of the proteolytic cleavage shows the cleavage sites as annontations in the protein sequence. The accompanying table lists all the fragments which are between 10 and 15 amino acids long.*

fragment parameters, you might have more annotations on the sequence than fragments in the table.

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.

8.11   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 8.43.
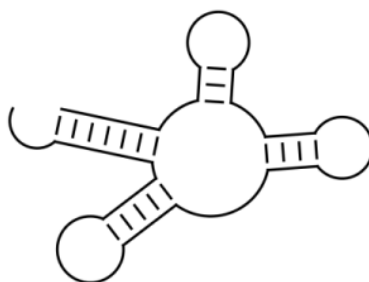
Figure 8.43: *Secondary structure of a tRNA molecule.*

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:

Download | 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 8.44).
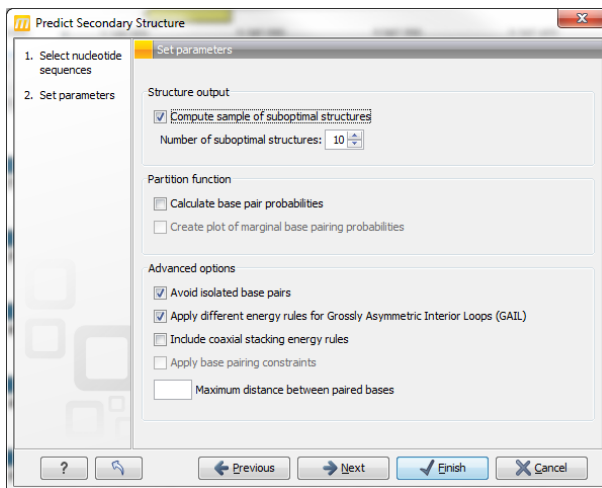
Figure 8.44: *Selecting to compute 10 suboptimal structures.*

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 8.45).
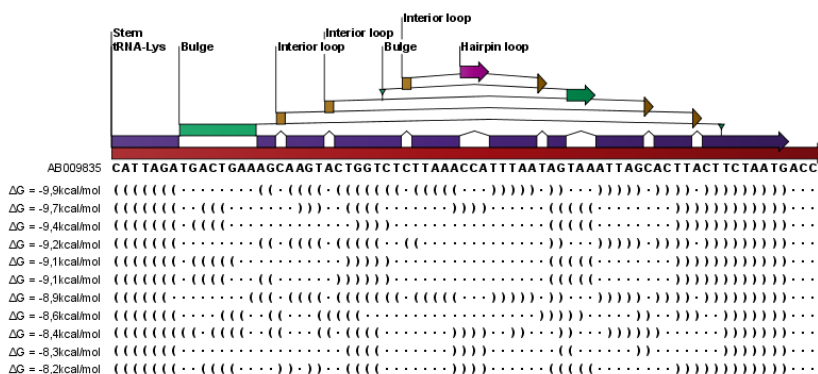
Figure 8.45: *The inital, linear view of the secondary structure prediction.*

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 8.46).
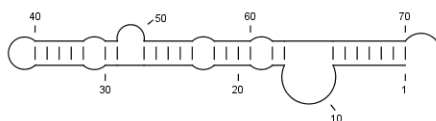
Figure 8.46: *The inital 2D view of the secondary structure.*

This structure does not look like the one we expected (shown in figure 8.43). 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** (![icon]):

Press and hold Ctrl (⌘ on Mac) | Show Secondary Structure Table (![icon])

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 8.47).
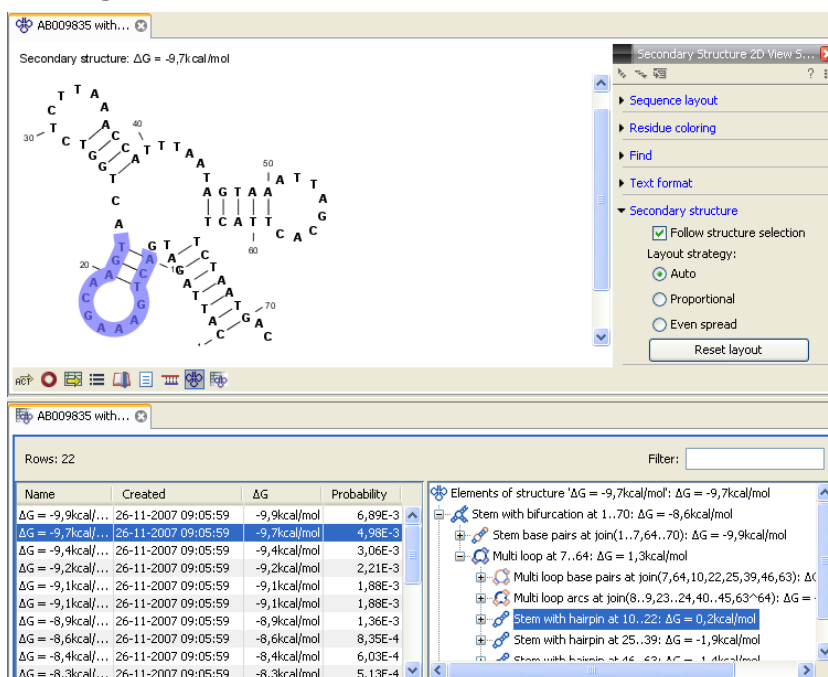
Figure 8.47: *A split view showing the scondary structure table at the bottom and the Secondary structure 2D view at the top. (You might need to Zoom out to see the structure).*

The secondary structure now looks very similar to figure 8.43. 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 (![icon]) the table view at the bottom.

Zoom in (![icon]) | 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** (![icon]) 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 31.14).

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 31.15, the structure shown in figure 31.14 has been modified by dragging with the
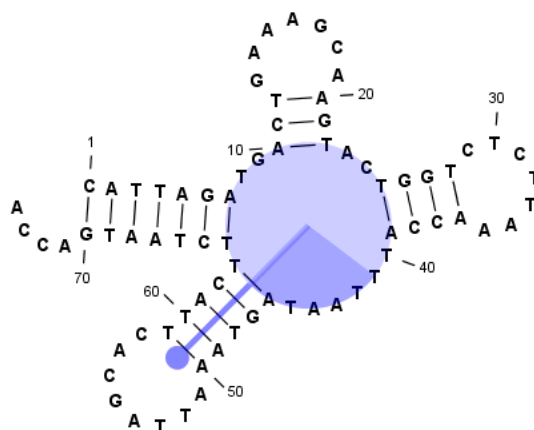
Figure 8.48: *The blue circle represents the anchor point for rotating the substructure.*

mouse.
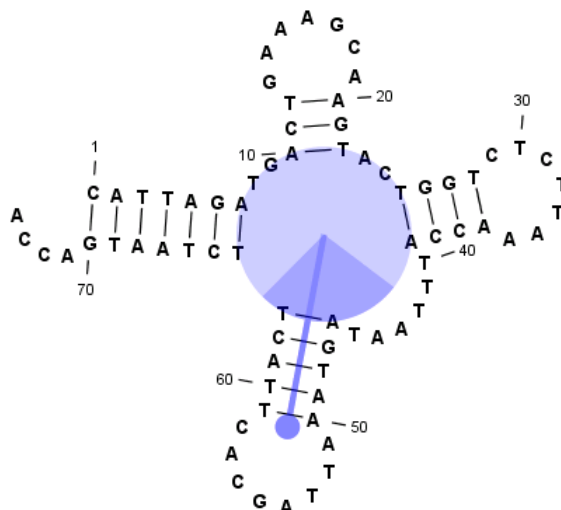
Figure 8.49: *The structure has now been rotated.*

The view can of course be printed ( ) or exported as graphics ( ).

8.12   Tutorial: Align Protein Sequences

This tutorial outlines some of the alignment functionality of the *CLC Main Workbench*. In addition to creating alignments of nucleotide or peptide sequences, the software offers several ways to view alignments. The alignments can then be used for building phylogenetic trees.

Sequences must be available via the **Navigation Area** to be included in an alignment. If you have sequences open in a View that you have not saved, then you just need to select the view tab and press Ctrl + S (or ⌘ + S on Mac) to save them.

In this tutorial six protein sequences from the Example data folder will be aligned. (See figure 8.50).

To align the sequences:
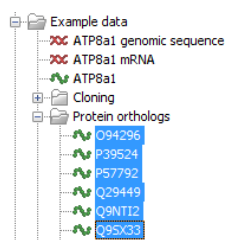
Figure 8.50: *Six protein sequences in 'Sequences' from the 'Protein orthologs' folder of the Example data.*

> **select and then right click on the sequences from the 'Protein orthologs' folder under 'Example Data' | Toolbox | Classical Sequence Analysis (🗂) | Alignments and Trees (📑)| Create Alignment (▦)**

8.12.1 The alignment dialog

This opens the dialog shown in figure 8.51.
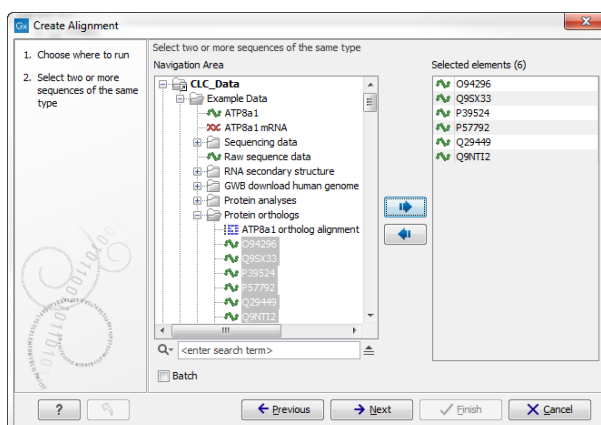
Figure 8.51: *The alignment dialog displaying the six protein sequences.*

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 8.52.
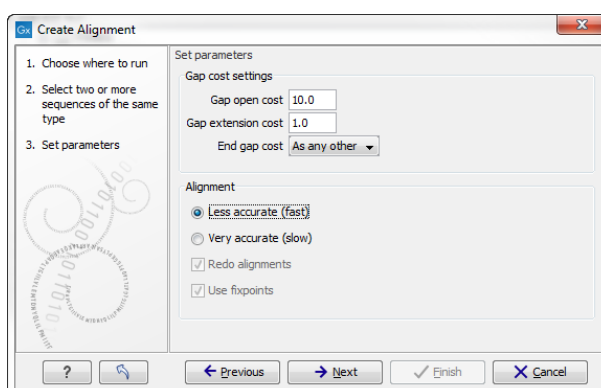
Figure 8.52: *The alignment dialog displaying the available parameters which can be adjusted.*

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. 8.53):
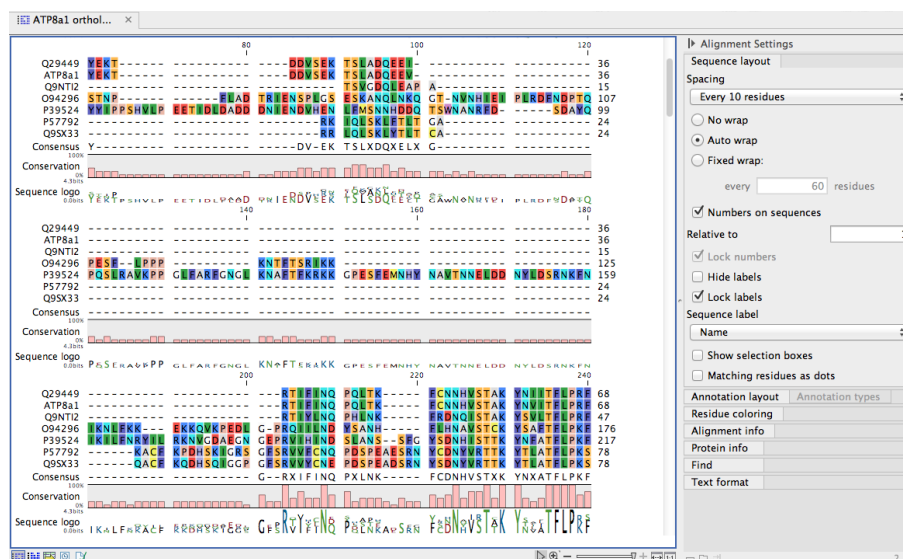
Figure 8.53: *The resulting alignment.*

Note! The new alignment is not saved automatically.

To save the alignment, drag the tab of the alignment view into the **Navigation Area**.

Installing the Additional Alignments plugin gives you access to two other alignment algorithms: ClustalW (Windows/Mac/Linux) and Muscle (Windows/Mac/Linux). The Additional Alignments Module can be downloaded from http://www.clcbio.com/plugins. Note that you will need administrative privileges on your system to install it.

8.13   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**.

8.13.1   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 8.54).

The restriction sites are shown on the sequence with an indication of cut site and recognition

Figure 8.54: *Showing restriction sites of ten restriction enzymes.*

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.

8.13.2 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 (figure 8.55).

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 8.56.
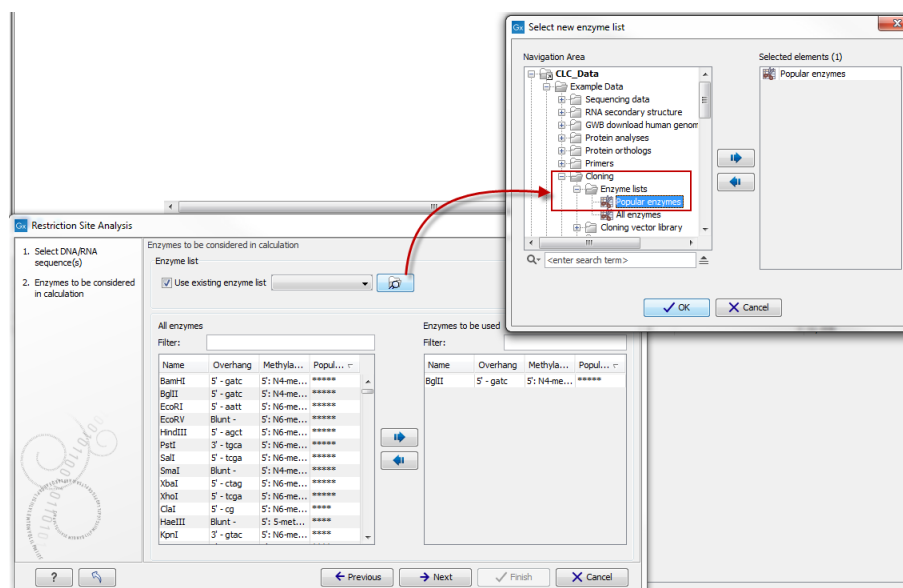
Figure 8.55: *Selecting enzyme list.*
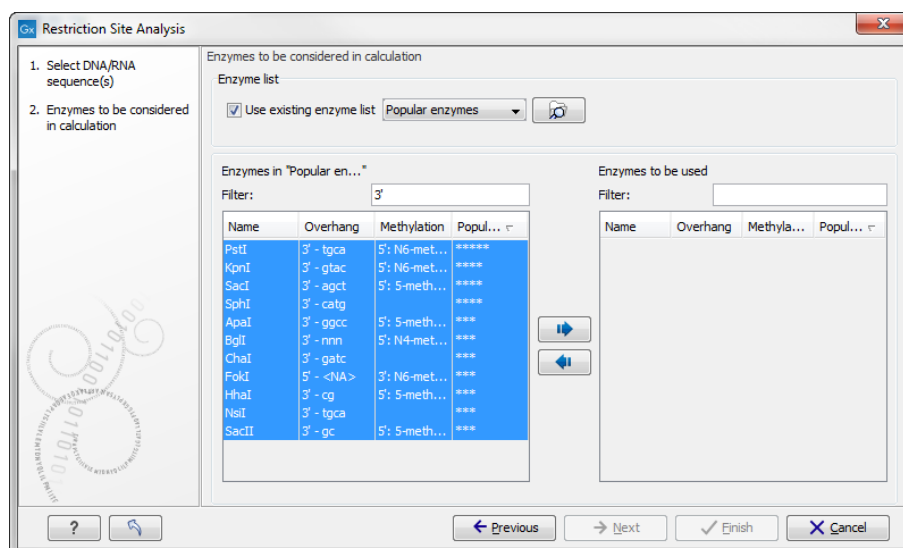
Figure 8.56: *Selecting enzymes.*

Click on the button labeled **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 (See figure 8.57).

Click on the button labeled **Next** and select that you want to **Add restriction sites as annotations on sequence** and **Create restriction map** (figure 8.58).

Click on the button labeled **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 8.59.

The restriction map at the bottom can also be shown as a table of fragments produced by cutting
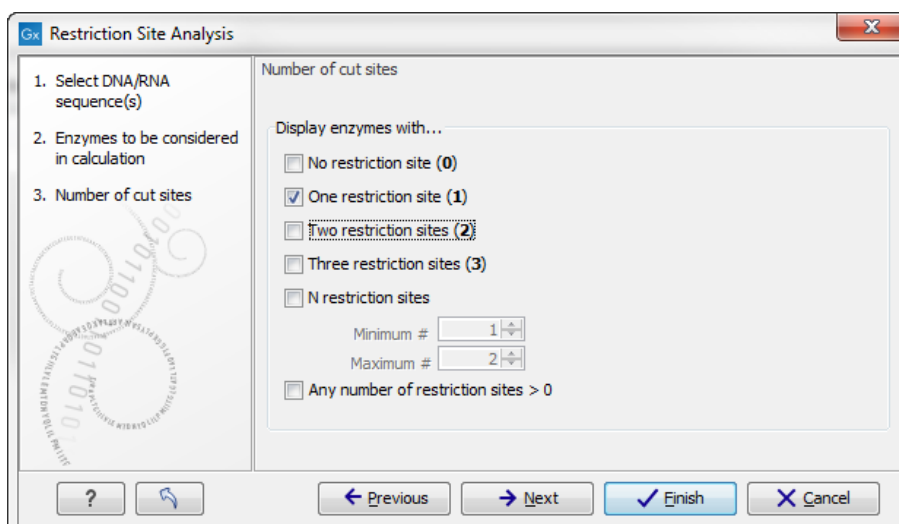
Figure 8.57: *Selecting output for restriction map analysis.*
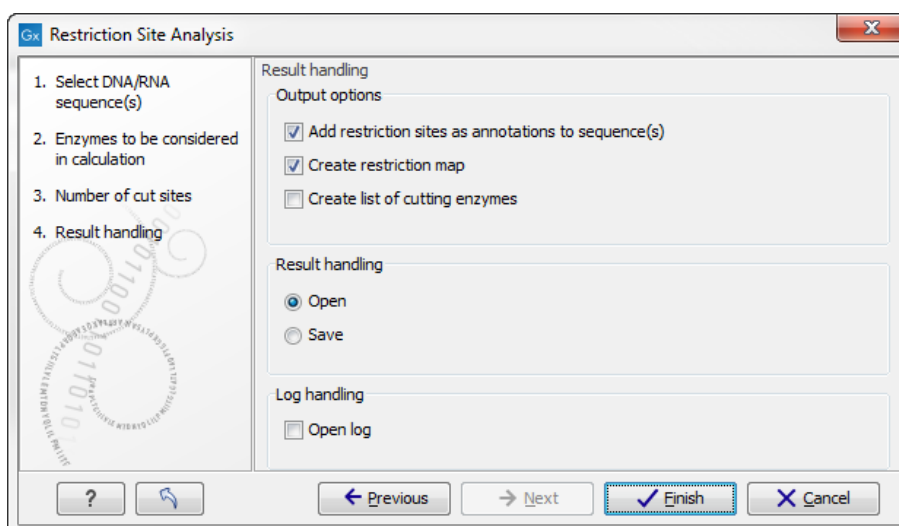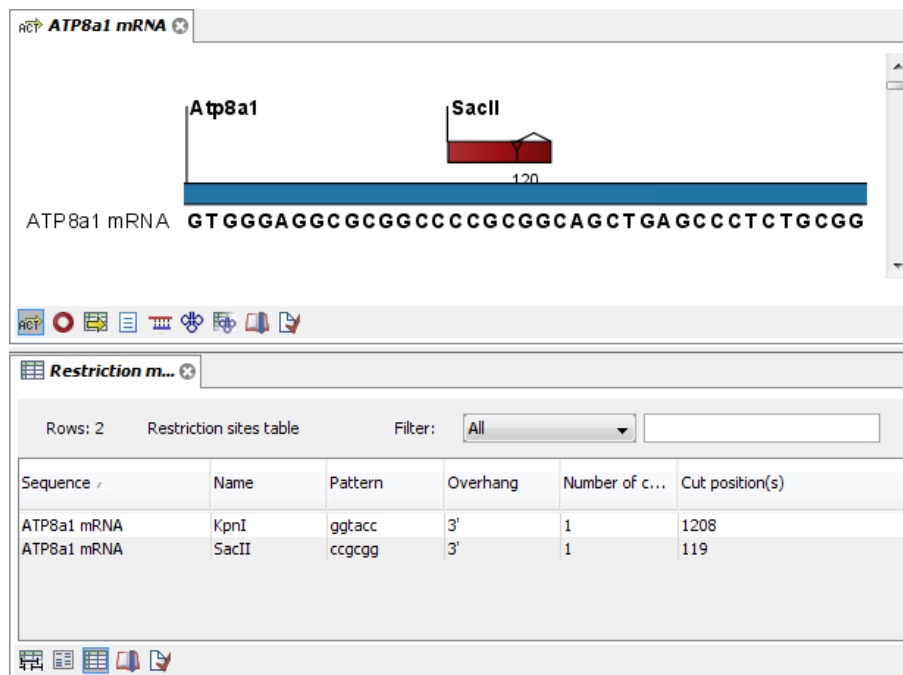
Figure 8.58: *Add restriction sites as annotations on sequence and create restriction map.*

the sequence with the enzymes:

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

Figure 8.59: *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.*

Part II

Core Functionalities

Chapter 9

User interface

Contents

This chapter provides an overview of the different areas in the user interface of *CLC Main Workbench*. As can be seen from figure 9.1 this includes a **Navigation Area**, **View Area**, **Menu Bar**, **Toolbar**, **Status Bar** and **Toolbox**.

A description of the **Navigation Area** is tightly connected to the data management features of *CLC Main Workbench* and can be found in section 10.1.
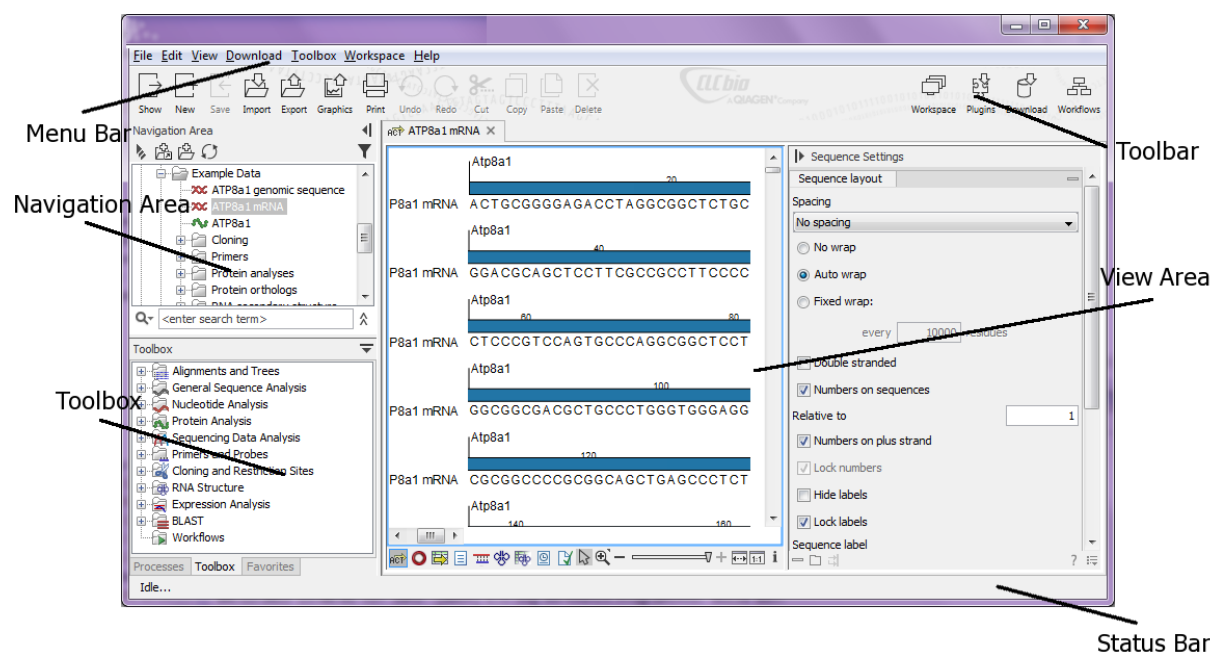
Figure 9.1: *The user interface consists of the Menu Bar, Toolbar, Status Bar, Navigation Area, Toolbox, and View Area.*

9.1   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 9.2.
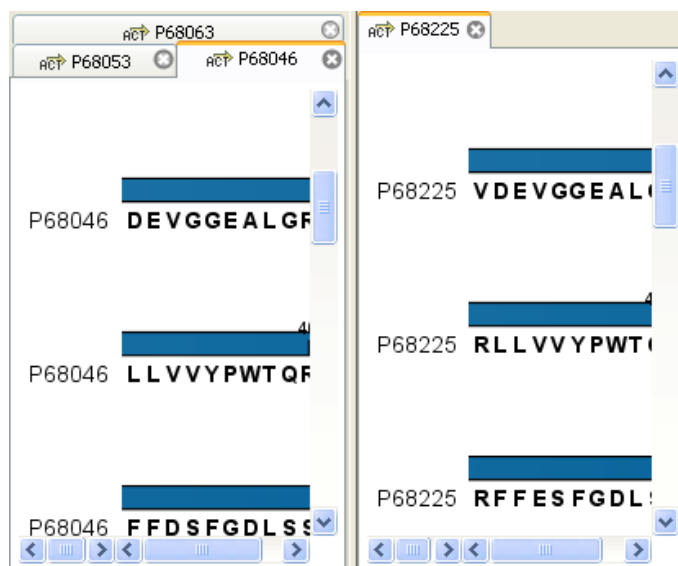
Figure 9.2: *A View Area can enclose several views, each view is indicated with a tab (see right view, which shows protein P68225). Furthermore, several views can be shown at the same time (in this example, four views are displayed).*

The tab concept is central to working with *CLC Main 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.

This chapter deals with the handling of views inside a **View Area**. Furthermore, it deals with rearranging the views.

Section 9.2 deals with the zooming and selecting functions.

9.1.1 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 10.1.5 for instructions on how to open a view using drag and drop.

9.1.2 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 9.3).

Figure 9.3: *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 (⬚), 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 17.5).

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 (⬤)**

9.1.3   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 9.4
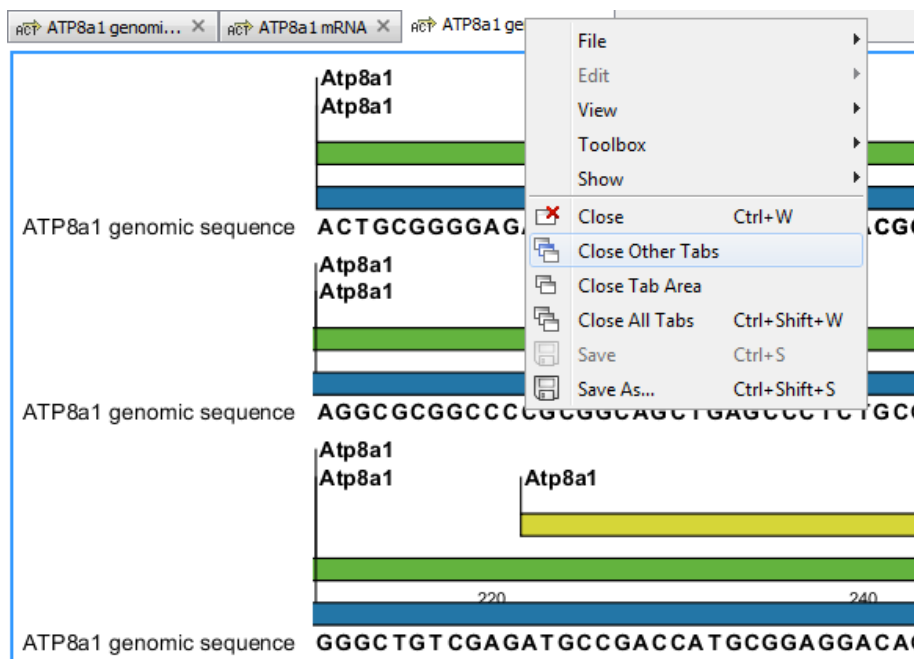
Figure 9.4: *By right-clicking a tab, several close options are available.*

- **Close.** See above.
- **Close Other Tabs.** Closes all other tabs, in all tab areas, except the one that is selected.
- **Close Tab Area.** Closes all tabs in the tab area.
- **Close All Tabs.** Closes all tabs, in all tab areas. Leaves an empty workspace.

9.1.4   Save changes in a view

When changes to an element 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 tab of a view containing an element that has been changed since you opened it, you are asked if you want to save.

When saving an element from 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 9.5).
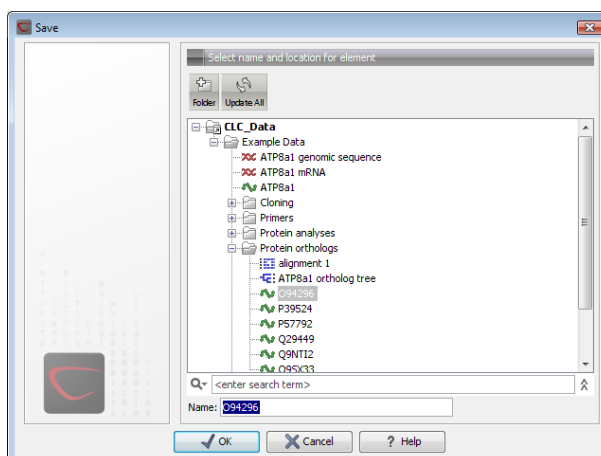
Figure 9.5: *Save dialog.*

In the dialog you select the folder in which you want to save the element.

After naming the element, press **OK**

9.1.5   Undo/Redo

If you make a change to an element 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 10.1.7).

You can set the number of possible undo actions in the Preferences dialog (see section 11).

9.1.6   Arrange views in View Area

To provide more space for viewing data, you can hide **Navigation Area** and the **Toolbox** by clicking the hide icon (◀|) at the top of the **Navigation 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. 9.6) illustrating that the view will be placed in this part of the **View Area**.
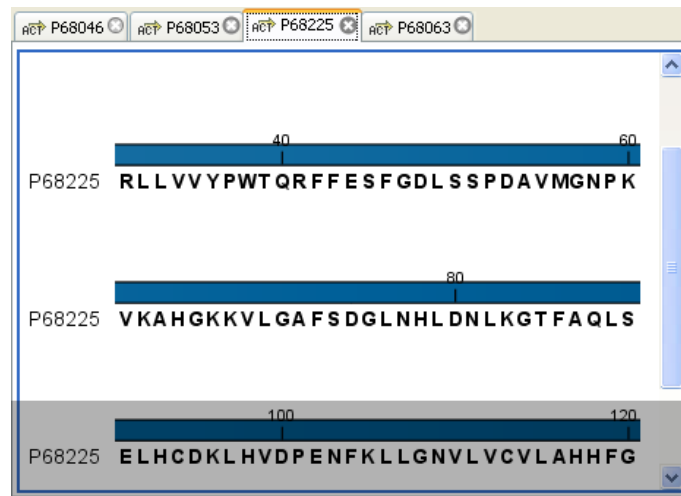
Figure 9.6: *When dragging a view, a gray area indicates where the view will be shown.*

The results of this action is illustrated in figure 9.7.

Figure 9.7: *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 horizontally 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 9.8). When the split is made vertically, the new view opens to the right of the existing view.

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.
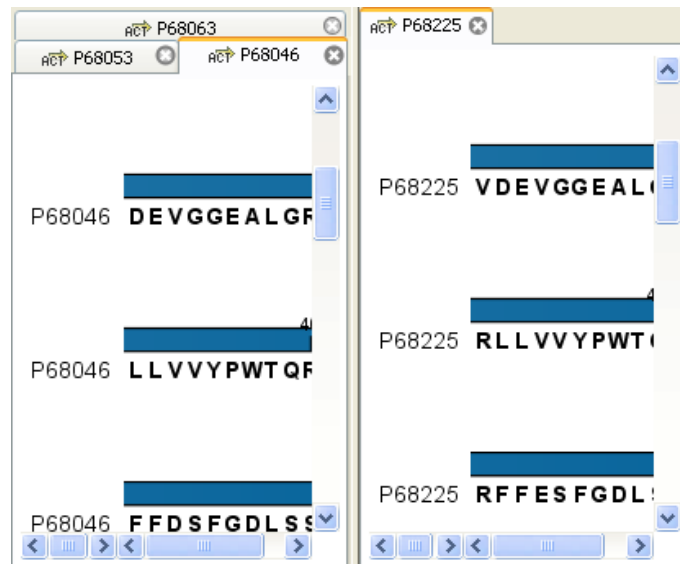
Figure 9.8: *A vertical split-screen.*

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.
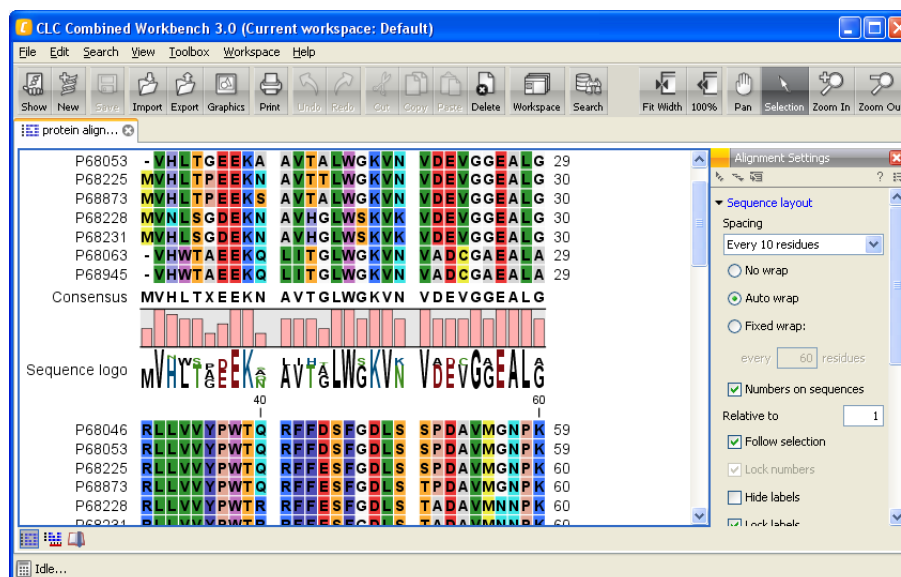
Figure 9.9: *A maximized view. The function hides the Navigation Area and the Toolbox.*

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**

Please note that you can also hide **Navigation Area** and the **Toolbox** by clicking the hide icon
(◀|) at the top of the **Navigation Area**

9.1.7 Moving a view to a different screen

Using multiple screens can be a great benefit when analyzing data with the *CLC Main Workbench*.
You can move a view to another screen by dragging the tab of the view and dropping it outside
the workbench window. Alternatively, you can right-click in the view area or on the tab itself and
select **View | Move to New Window** from the context menu.

An example is shown in figure 9.10, where the main Workbench window shows a table of open
reading frames, and the screen to the right is used to display the sequence and annotations.
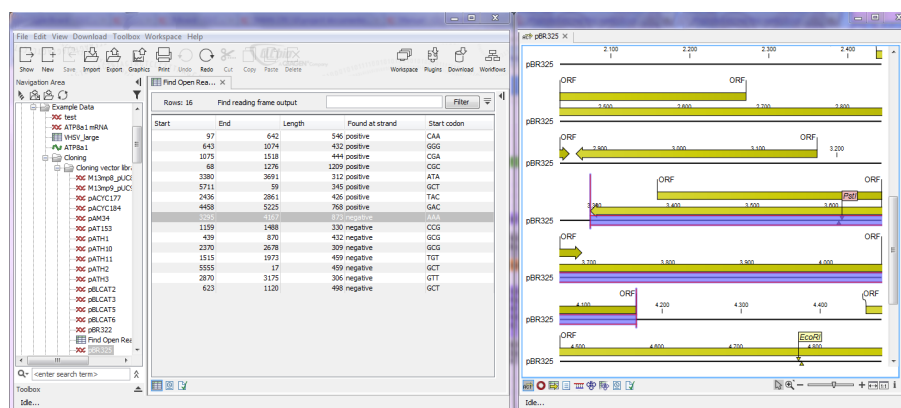
Figure 9.10: *Showing the table on one screen while the sequence is displayed on another screen.
Clicking the table of open reading frames causes the view on the other screen to follow the
selection. Note that the screen resolution in this figure is kept low in order to include it in the
manual; in a real scenario, the resolution will be much higher.*

You can make more detached windows, by dropping tabs outside the open workbench windows,
or you can drag more tabs to a detached window. To get a tab back to the main workbench
window, just drag the detached tab back, and drop it next to the other tabs in the top of the view
area. **Note:** You should not drag the detached window header, just the tab itself.

You can also split the view area in the detached windows as described in section 9.1.6.

9.1.8 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.

Figure 9.11 shows the default **Side Panel** for a protein sequence. It is organized into *palettes*.

In this example, there is one for Sequence layout, one for Annotation Layout etc. These palettes
can be re-organized by dragging the palette name with the mouse and dropping it where you want
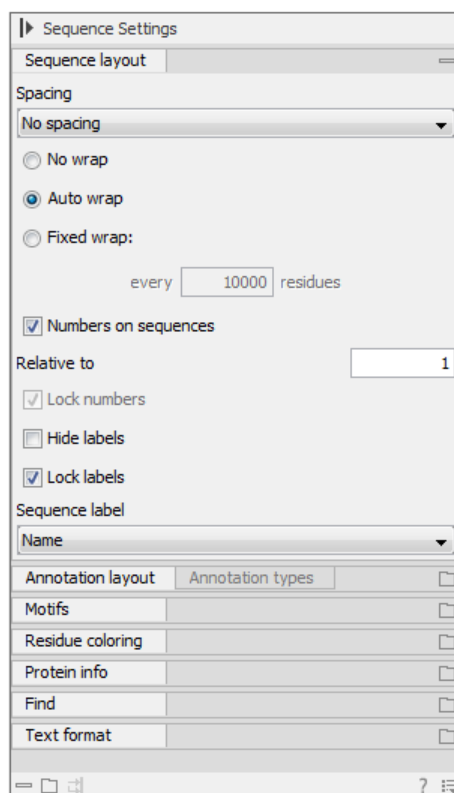
Figure 9.11: *The default view of the Side Panel when opening a protein sequence.*

it to be. They can either be situated next to each other, so that you can switch between them, or they can be listed on top of each other, so that expanding one of the palettes will push the palettes below further down.

In addition, they can be moved away from the **Side Panel** and placed anywhere on the screen as shown in figure 9.12.

In this example, the **Motifs** palette has been placed on top of the sequence view together with the **Protein info** and the **Residue coloring** palettes. In the **Side Panel** to the right, the **Find** palette has been put on top.

In order to make all palettes dock in the **Side Panel** again, click the **Dock Side Panel** icon (→|).

You can completely hide the **Side Panel** by clicking the **Hide Side Panel** icon (|▶).

At the bottom of the **Side Panel** (see figure 9.13) there are a number of icons used to:

- Expand all settings (▢).

- Collapse all settings (▭).

- Dock all palettes (⊐|)

- Get **Help** for the particular view and settings (?)

- Save the settings of the **Side Panel** or apply already saved settings. Read more in section 11.6
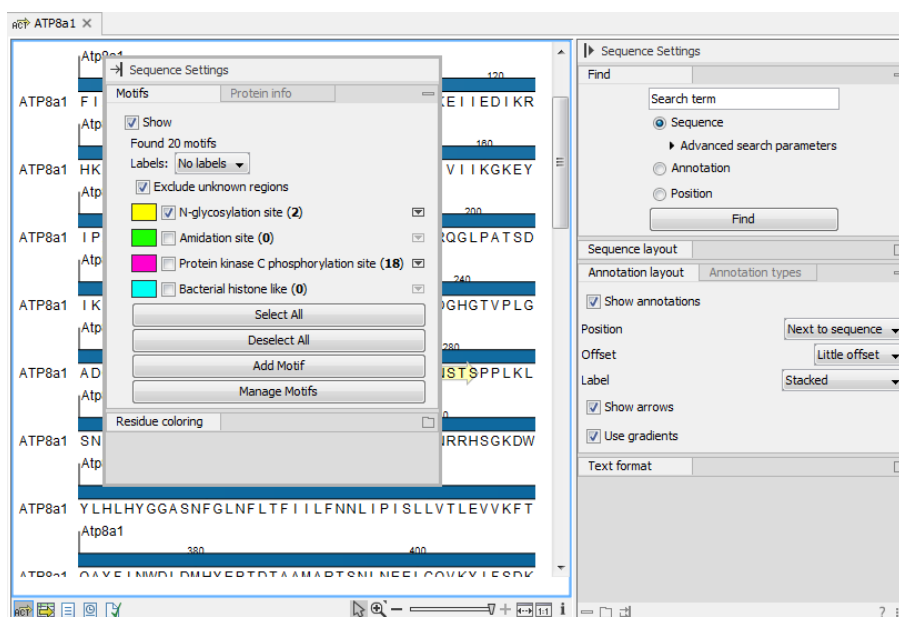
Figure 9.12: *Palettes can be organized in the Side Panel as you like or placed anywhere on the screen.*
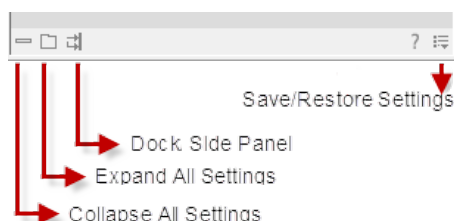
Figure 9.13: *Controlling the Side Panel at the bottom*

Note! Changes made to the **Side Panel**, including the organization of palettes will not be saved when you save the view. See how to save the changes in section 11.6

9.2 Zoom and selection in View Area

All views except tabular and text views support zooming. Figure 9.14 shows the zoom tools, located at the bottom right corner of the view.
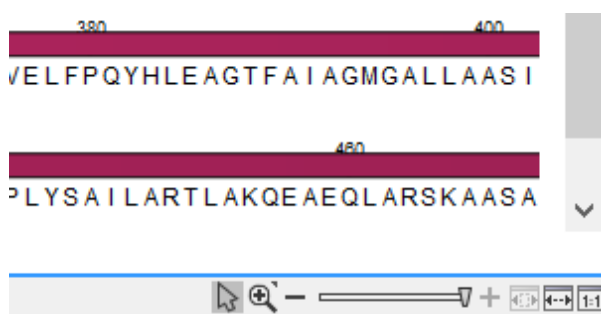
Figure 9.14: *The zoom tools are located at the bottom right corner of the view.*

The zoom tools consist of some shortcuts for zooming to fit the width of the view (⬌), zoom to 100 % to see details (1:1), zoom to a selection (⬌), a zoom slider, and two mouse mode

buttons ( ) ( ).

The slider reflects the current zoom level and can be used to quickly adjust this. For more fine-grained control of the zoom level, move the mouse upwards while sliding.

The sections below describes how to use these tools as well as other ways of zooming and navigating data.

Please note that when working with protein 3D structures, there are specific ways of controlling zooming and navigation as explained in section .

9.2.1   Zoom in

There are six ways of **Zooming In**:

> **Click Zoom In ( ) in the zoom tools (or press Ctrl+2) | click the location in the.
> view that you want to zoom in on**

> or **Click Zoom In ( ) in the zoom tools | 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**

> or **Move the zoom slider located in the zoom tools**

> or **Click the plus icon in the zoom tools**

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**

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 in zoom mode, the zoom function is reversed.

If you want to zoom in to 100 % to see all the data, click the **Zoom to Max** ( ) icon.

9.2.2   Zoom out

It is possible to zoom out in different ways:

> **Click Zoom Out ( ) in the zoom tools (or press Ctrl+3) | click in the view**

> or **Press '-' on your keyboard**

> or **Move the zoom slider located in the zoom tools**

> or **Click the minus icon in the zoom tools**

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**

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 zoom out to see all the data, click the **Zoom to Fit** ( ) icon.

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.

9.2.3   Selecting, panning and zooming

In the zoom tools, you can control which mouse mode to use. The default is **Selection mode** ( ) which is used for selecting data in a view. Next to the selection mode, you can select the **Zoom in mode** as described in section 9.2.1. If you press and hold this button, two other modes become available as shown in figure 9.15:

- **Panning** ( ) is used for dragging the view with the mouse as a way of scrolling.

- **Zoom out** ( ) is used to change the mouse mode so that whenever you click the view, it zooms out.
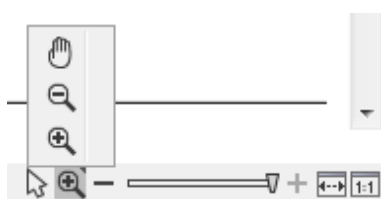
Figure 9.15: *Additional mouse modes can be found in the zoom tools.*

If you hold the mouse over the selection and zoom tools, tooltips will appear that provide further information about how to use the tools.

The mouse modes only apply when the mouse is within the view where they are selected.

The **Selection mode** can also be invoked with the keyboard shortcut Ctrl+1, while the **Panning mode** can be invoked with Ctrl+4.

For some views, if you have made a selection, there is a **Zoom to Selection** ( ) button, which allows you to zoom and scroll directly to fit the view to the selection.

9.3   Toolbox and Status Bar

The **Toolbox** is placed in the left side of the user interface of *CLC Main Workbench* below the **Navigation Area**.

The **Toolbox** shows a **Processes tab**, **Favorites tab** and a **Toolbox tab**.

The **Toolbox** can be hidden, so that the **Navigation Area** is enlarged and thereby displays more elements:

>   **View** | **Show/Hide Toolbox** | **Show/Hide Toolbox**

You can also click the **Hide Toolbox** ( ) button.

9.3.1   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 9.16).

Running and paused processes are not deleted.
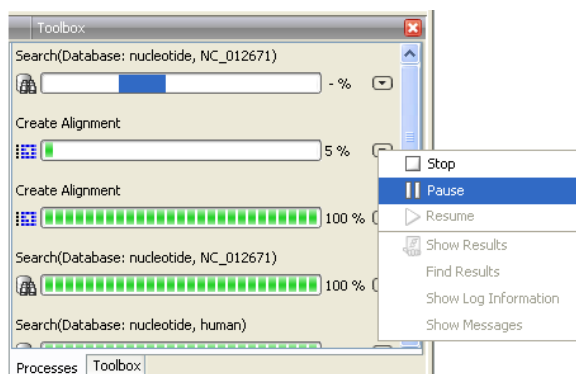
Figure 9.16: *A database search and an alignment calculation are running. Clicking the small icon next to the process allow you to stop, pause and resume processes.*

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 15.2), 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 15.2), 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 Finished 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.

9.3.2   Toolbox

The content of the **Toolbox** tab in the **Toolbox** corresponds to **Toolbox** in the **Menu Bar**.

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**.

Quick access to tools

To enable quick launch of tools in *CLC Main Workbench*, press Ctrl + Shift + T (⌘ + Shift + T on Mac) to show the quick launch dialog (see figure 9.17).
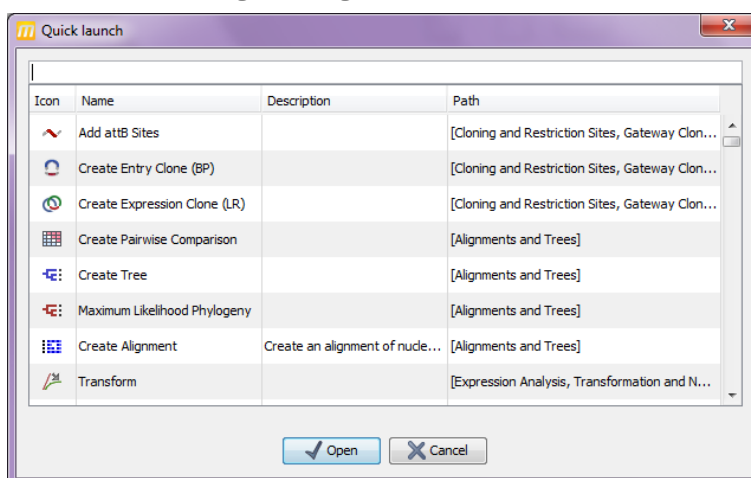
Figure 9.17: *Quick access to all tools in* **CLC Main Workbench**.

When the dialog is opened, you can start typing search text in the text field at the top. This will bring up the list of tools that match this text either in the name, description or location in the Toolbox. In the example shown in figure 9.18, typing `plot` shows a list of tools involving plots, and the arrow keys or mouse can be used for selecting and starting a tool.
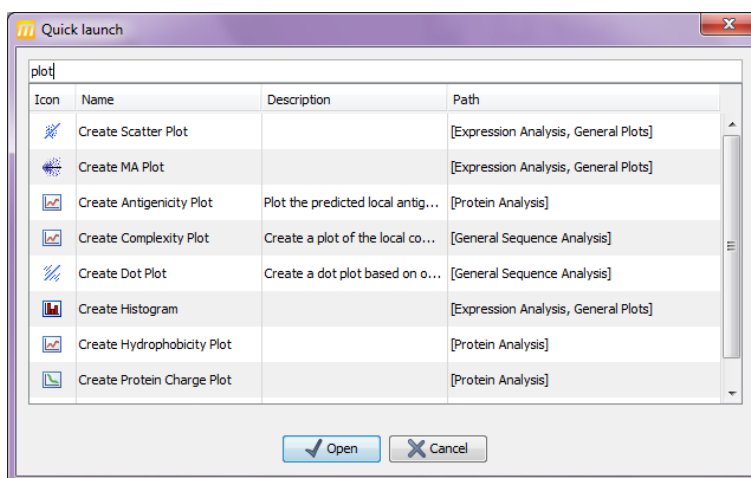
Figure 9.18: *Typing in the search field at the top will filter the list of tools to launch.*

Favorites toolbox

Next to the **Toolbox** tab, you find the **Favorites** tab. This can be used for organizing and getting quick access to the tools you use the most. It consists of two parts as shown in figure 9.19.

Favorites You can manually add tools to the favorites menu simply by right-clicking the tool in the Toolbox. You can also right-click the Favorites folder itself and select **Add Tool**. To remove a tool, right-click and select **Remove from Favorites**. Note that you can also add complete folders to the favorites.
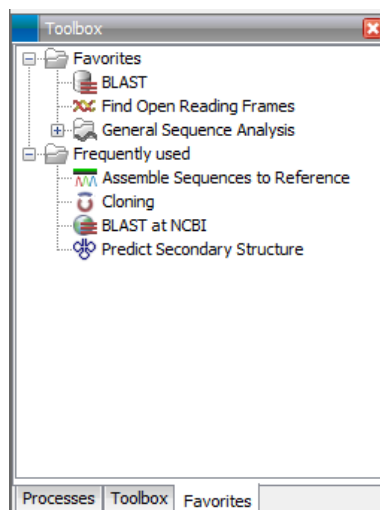
Figure 9.19: *Favorites toolbox.*

Frequently used The list of tools in this folder is automatically populated as you use the Workbench. The most frequently used tools are listed at the top.

9.3.3   Status Bar

As can be seen from figure 9.1, 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 9.2.3 for more about the Selection mode button.)

9.4   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 Main 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 Main Workbench* at a time. Use two or more **Workspaces** instead.

9.4.1   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 Main 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 9.20).
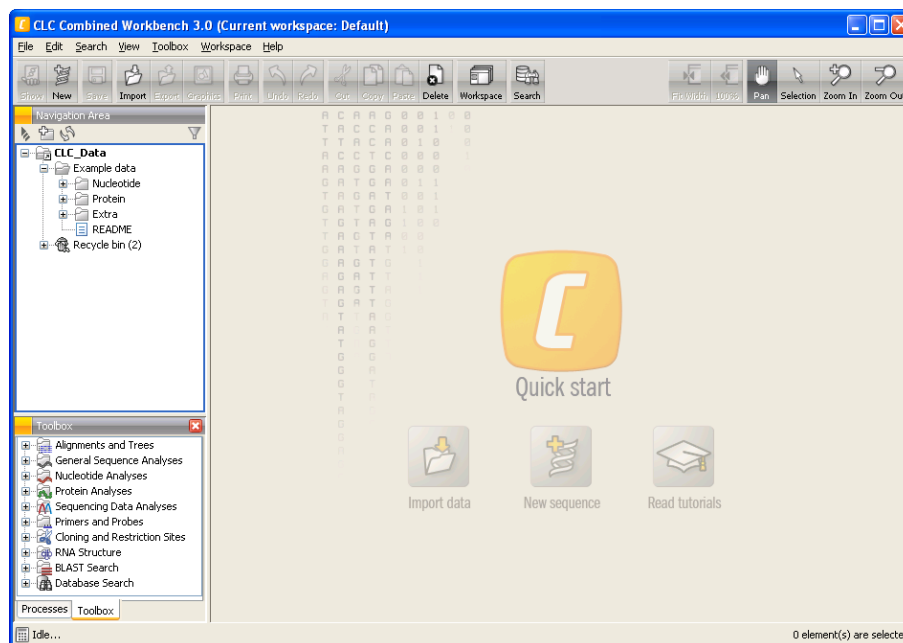
Figure 9.20: *An empty Workspace.*

9.4.2   Select Workspace

When there is more than one **Workspace** in the *CLC Main 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 Main Workbench*" at the top left corner of the main window, in figure 9.20 it says: (default).

9.4.3   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.

9.5   List of shortcuts

The keyboard shortcuts in *CLC Main Workbench* are listed below.

Action	Windows/Linux	Mac OS X
Adjust selection	Shift + arrow keys	Shift + arrow keys
Adjust workflow layout	Shift + Alt + L	⌘ + Shift + Alt + L
Close	Ctrl + W	⌘ + W
Close all views	Ctrl + Shift + W	⌘ + Shift + W
Copy	Ctrl + C	⌘ + C
Create alignment	Ctrl + A	⌘ Shift + A
Create track list	Ctrl + L	⌘ + L
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	'.' (dot)	'.' (dot)
Find Previous Conflict	',' (comma)	',' (comma)
Help	F1	F1
Import	Ctrl + I	⌘ + I
Maximize/restore size of View	Ctrl + M	⌘ + M
Move gaps in alignment	Ctrl + arrow keys	⌘ + arrow keys
New Folder	Ctrl + Shift + N	⌘ + Shift + N
New Sequence	Ctrl + N	⌘ + N
Panning Mode	Ctrl + 4	⌘ + 4
Paste	Ctrl + V	⌘ + V
Print	Ctrl + P	⌘ + P
Redo	Ctrl + Y	⌘ + Y
Rename	F2	F2
Reverse Complement	Ctrl + R	⌘ + R
Save	Ctrl + S	⌘ + S
Scrolling horizontally	Shift + Scroll wheel	Shift + Scroll wheel
Search local data	Ctrl + Shift + F	⌘ + Shift + F
Search via Side Panel	Ctrl + F	⌘ + F
Search NCBI	Ctrl + B	⌘ + B
Search UniProt	Ctrl + Shift + U	⌘ + Shift + U
Select All	Ctrl + A	⌘ + A
Select Selection Mode	Ctrl + 1	⌘ + 1
Show folder content	Ctrl + O	⌘ + O
Show/hide Side Panel	Ctrl + U	⌘ + U
Sort folder	Ctrl + Shift + R	⌘ + Shift + R
Split Horizontally	Ctrl + T	⌘ + T
Split Vertically	Ctrl + J	⌘ + J
Start Tool Quick Launch	Ctrl + Shift + T	⌘ + Shift + T
Translate to Protein	Ctrl + Shift + P	⌘ + Shift + P
Undo	Ctrl + Z	⌘ + Z
Update folder	F5	F5
User Preferences	Ctrl + K	⌘ + ;
Vertical scroll in read tracks	Alt + Scroll wheel	Alt + Scroll wheel
Vertical scroll in reads tracks, fast	Shift+Alt+Scroll wheel	Shift+Alt+Scroll wheel
Vertical zoom in graph tracks	Alt + Scroll wheel	Alt + Scroll wheel

Action	Windows/Linux	Mac OS X
Reverse zoom mode	press and hold Shift	press and hold Shift
Workflow, add element	Alt + Shift + E	Alt + Shift + E
Workflow, collapse if its expanded	Alt + Shift + '-' (minus)	Alt + Shift + '-'
Workflow, create installer	Alt + Shift + I	Alt + Shift + I
Workflow, execute	Ctrl + enter	⌘ + enter
Workflow, expand if its collapsed	Alt + Shift + '+' (plus)	Alt + Shift + '-'
Workflow, highlight used elements	Alt + Shift + U	Alt + Shift + U
Workflow, remove all elements	Alt + Shift + R	Alt + Shift + R
Zoom	Ctrl + Scroll wheel	Ctrl + Scroll wheel
Zoom In Mode	Ctrl + 2	⌘ + 2
Zoom In (without clicking)	'+' (plus)	'+' (plus)
Zoom Out Mode	Ctrl + 3	⌘ + 3
Zoom Out (without clicking)	'-' (minus)	'-' (minus)

Combinations of keys and mouse movements are listed below.

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 mode	Shift	Shift	Click in view
Select multiple elements that are not grouped together	Ctrl	⌘	Click elements
Select multiple elements that are grouped together	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 10

Data management and search

Contents

This chapter explains the data management features of *CLC Main Workbench*. The first section explains the basics of the data organization and the **Navigation Area**. The next section explains how to set up custom attributes for the data that can be used for more advanced data management. Finally, there is a section about how to search through local data.

10.1 Navigation Area

The **Navigation Area** is located in the left side of the screen, under the **Toolbar** (see figure 10.1). It is used for organizing and navigating data. Its behavior is similar to the way files and folders are usually displayed on your computer.
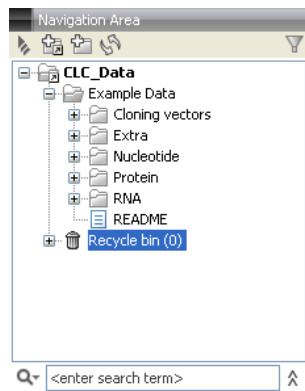
Figure 10.1: *The Navigation Area.*

To provide more space for viewing data, you can hide **Navigation Area** and the **Toolbox** by clicking the hide icon (◀|) at the top.

10.1.1 Data structure

The data in the **Navigation Area** is organized into a number of **Locations**. When the *CLC Main 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 10.2. The full path to the system folder can be located by mousing over the data location as shown in figure 10.3.

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 10.4).
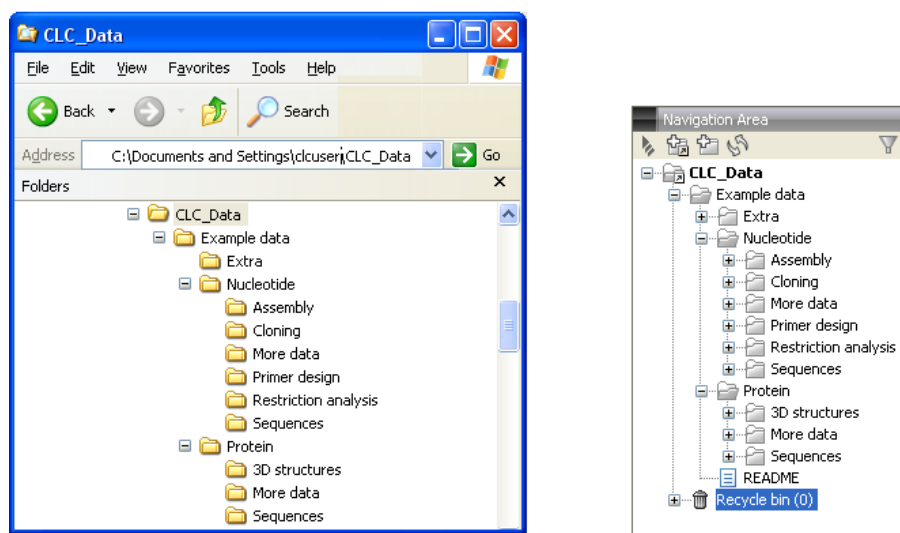
Figure 10.2: *In this example the location called 'CLC_Data' points to the folder at C:\Documents and settings\clcuser\CLC_Data.*
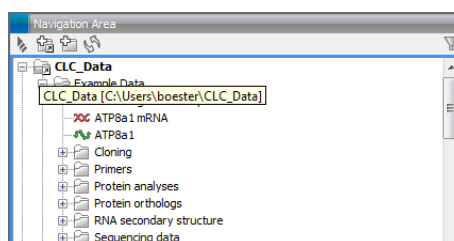
Figure 10.3: *Mousing over the location called 'CLC_Data' shows the full path to the system folder, which in this case is C:\Users\boester\CLC_Data.*
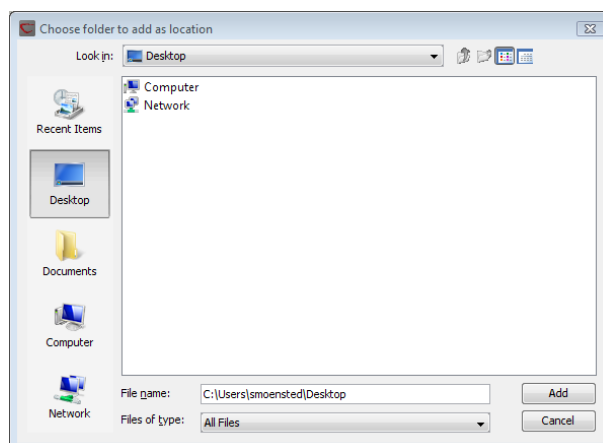
Figure 10.4: *Navigating to a folder to use as a new location.*

When you click **Open**, the new location is added to the **Navigation Area** as shown in figure 10.5.

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 location will appear *inactive* when you are not connected. Once you connect to the drive again,
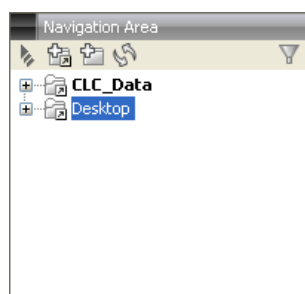
Figure 10.5: *The new location has been added.*

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 9.1.

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 13). 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 10.1.1).

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.

10.1.2   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.

10.1.3 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.

10.1.4 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.

10.1.5 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 "grayed 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 9.1.6).

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**

10.1.6 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 two different ways:

> **select the element | Edit in the Menu Bar | Rename**

> or **select the element | F2**

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 17.3.4.

10.1.7 Delete, restore and remove elements

When one deletes data from a data folder in the Workbench, it is moved to the recycle bin in that data location. Each data location has its own recycle bin. From the recycle bin, data can then be restored, or completely removed. Removal of data from the recycle bin frees disk space.

Deleting a folder or an element from a Workbench data location 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 or until you choose to restore the data object to your data location.

For deleting annotations instead of folders or elements, see section 17.3.5.

Items in a recycle bin can be restored in two ways:

> Drag the elements with the mouse into the folder where they used to be.

or **select the element | right click and choose the option Restore**.

Once restored, you can continue to work with that data.

All contents of the recycle bin can be removed by choosing to empty the recycle bin:

> **Edit in the Menu Bar | Empty Recycle Bin (🗑)**

This deletes the data and frees up disk space.

Note! This cannot be undone. Data is not recoverable after it is removed by emptying the recycle bin.

10.1.8   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**

or

> **select a folder or location | right click on the folder and select Show (▱) | Contents (📁)**

An example is shown in figure 10.6.
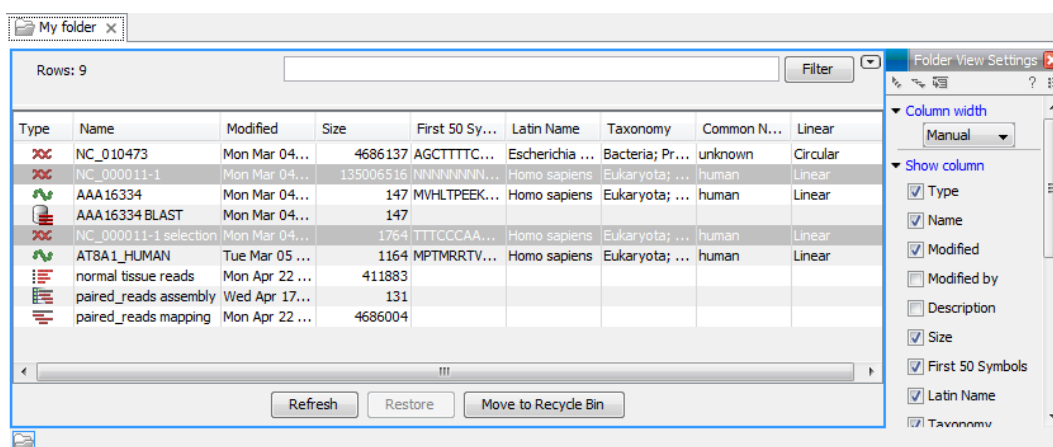
Figure 10.6: *Viewing the elements in a folder.*

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.

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 for example the description or name of several elements in one go.

In figure 10.7 you can see an example where the name of two sequence are renamed in one go. In this example, a dialog with a text field will be shown, letting you enter a new name for these two sequences.
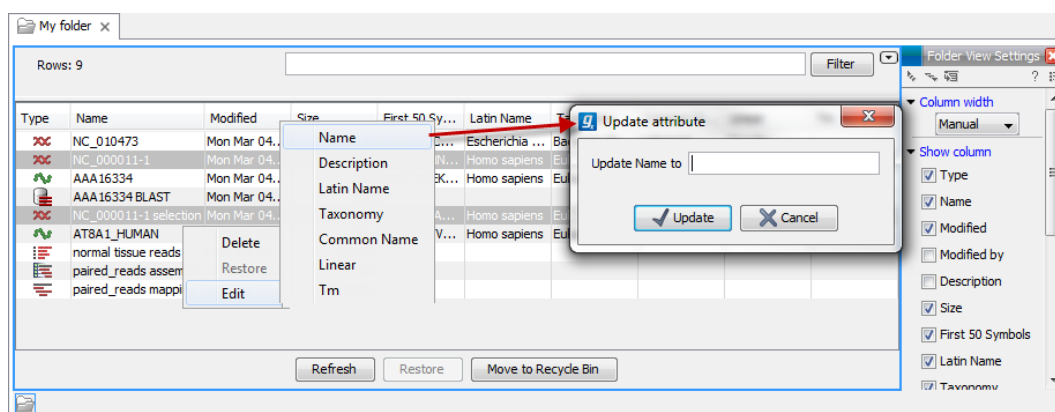
Figure 10.7: *Changing the common name of two sequences.*

Note! This information is directly saved and you cannot undo.

Drag and drop folder elements

You can drag and drop objects from the folder editor to the **Navigation area**. This will create a copy of the objects at the selected destination. New elements can be included in the folder editor in the view area by dragging and dropping an element from a destination in the **Navigation Area** to the folder in the **Navigation Area** that you have open in the view area. It is not possible to drag elements directly from the **Navigation Area** to the folder editor in the View area.

10.2   Customized attributes on data locations

The *CLC Main Workbench* makes it possible to define location-specific attributes on all elements stored in a data location. This could be company-specific information such as LIMS id, freezer position etc. Note that the attributes scheme belongs to a location, so if you have added multiple locations, they will have their own separate set of attributes.

Note! A Metadata Import Plugin is available. The plugin consists of two tools: "Import Sequences in Table Format" and "Associate with metadata". These tools allow sequences to be imported

from a tabular data source and make it possible to add metadata to existing objects.

10.2.1   Configuring which fields should be available

To configure which fields that should be available[1]:

right-click the data location | Location | Attribute Manager

This will display the dialog shown in figure 10.8.
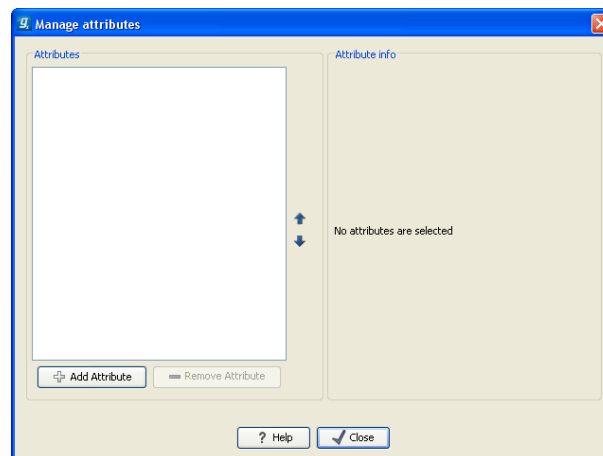
Figure 10.8: *Adding attributes.*

Click the **Add Attribute** ( ) button to create a new attribute. This will display the dialog shown in figure 10.9.
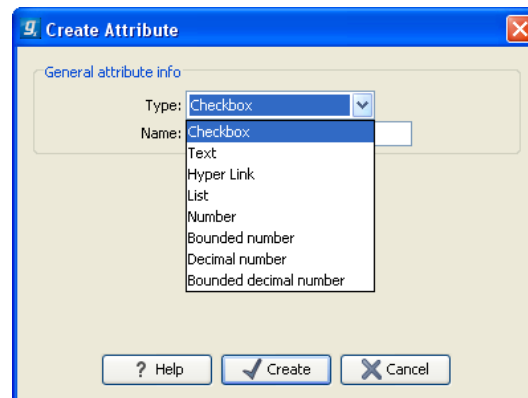
Figure 10.9: *The list of attribute types.*

First, select what kind of attribute you wish to create. This affects the type of information that can be entered by the end users, and it also affects the way the data can be searched. The following types are available:

- **Checkbox**. This is used for attributes that are binary (e.g. true/false, checked/unchecked and yes/no).

- **Text**. For simple text with no constraints on what can be entered.

---

[1]If the data location is a server location, you need to be a server administrator to do this

- **Hyper Link**. This can be used if the attribute is a reference to a web page. A value of this type will appear to the end user as a hyper link that can be clicked. Note that this attribute can only contain one hyper link. If you need more, you will have to create additional attributes.

- **List**. Lets you define a list of items that can be selected (explained in further detail below).

- **Number**. Any positive or negative integer.

- **Bounded number**. Same as number, but you can define the minimum and maximum values that should be accepted. If you designate some kind of ID to your sequences, you can use the bounded number to define that it should be at least 1 and max 99999 if that is the range of your IDs.

- **Decimal number**. Same as number, but it will also accept decimal numbers.

- **Bounded decimal number**. Same as bounded number, but it will also accept decimal numbers.

When you click **OK**, the attribute will appear in the list to the left. Clicking the attribute will allow you to see information on its type in the panel to the right.

10.2.2 Editing lists

Lists are a little special, since you have to define the items in the list. When you click a list in the left side of the dialog, you can define the items of the list in the panel to the right by clicking **Add Item** (➕) (see figure 10.10).
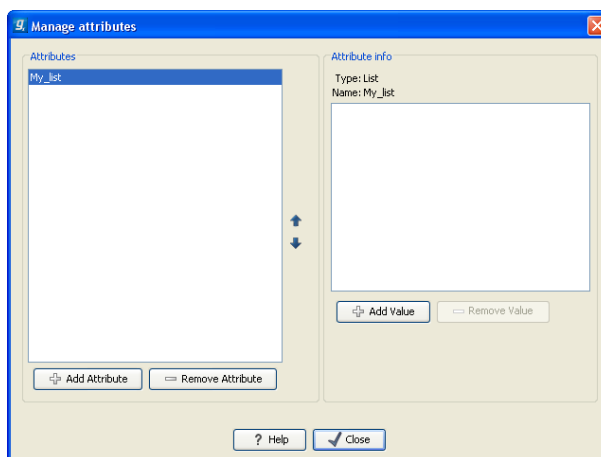
Figure 10.10: *Defining items in a list.*

Remove items in the list by pressing **Remove Item** (➖).

10.2.3 Removing attributes

To remove an attribute, select the attribute in the list and click **Remove Attribute** (➖). This can be done without any further implications if the attribute has just been created, but if you remove an attribute where values have already been given for elements in the data location, it will have implications for these elements: The values will not be removed, but they will become static,

which means that they cannot be edited anymore. They can only be removed (see more about how this looks in the user interface below).

If you accidentally removed an attribute and wish to restore it, this can be done by creating a new attribute of exactly the same name and type as the one you removed. All the "static" values will now become editable again.

When you remove an attribute, it will no longer be possible to search for it, even if there is "static" information on elements in the data location.

Renaming and changing the type of an attribute is not possible - you will have to create a new one.

10.2.4   Changing the order of the attributes

You can change the order of the attributes by selecting an attribute and click the **Up** and **Down** arrows in the dialog. This will affect the way the attributes are presented for the user as described below.

10.3   Filling in values

When a set of attributes has been created (as shown in figure 10.11), the end users can start filling in information.
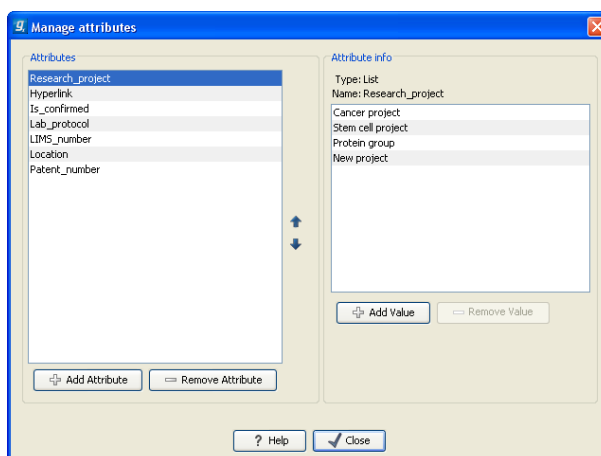
Figure 10.11: *A set of attributes defined in the attribute manager.*

This is done in the element info view:

> **right-click a sequence or another element in the Navigation Area | Show (⬒) | Element info (☑)**

This will open a view similar to the one shown in figure 10.12.

You can now enter the appropriate information and **Save**. When you have saved the information, you will be able to search for it (see below). Note that the sequence needs to be saved in the data location before you can edit the attribute values.

When nobody has entered information, the attribute will have a "Not set" written in red next to the attribute (see figure 10.13).

Figure 10.12: *Adding values to the attributes.*

Figure 10.13: *An attribute which has not been set.*

This is particularly useful for attribute types like checkboxes and lists where you cannot tell, from the displayed value, if it has been set or not. Note that when an attribute has not been set, you cannot search for it, even if it looks like it has a value. In figure 10.13, you will *not* be able to find this sequence if you search for research projects with the value "Cancer project", because it has not been set. To set it, simply click in the list and you will see the red "Not set" disappear.

If you wish to reset the information that has been entered for an attribute, press "Clear" (written in blue next to the attribute). This will return it to the "Not set" state.

The **Folder editor** provides a quick way of changing the attributes of many elements in one go (see section 10.1.8).

10.3.1   What happens when the sequence is copied to another data location?

The user supplied information, which has been entered in the **Element info**, is attached to the attributes that have been defined in this particular data location. If you copy the sequence to another data location or to a data location containing another attribute set, the information will become fixed, meaning that it is no longer editable and cannot be searched for. Note that attributes that were "Not set" will disappear when you copy data to another location.

If the sequence is moved back to the original data location, the information will again be editable

and searchable.

10.3.2 Searching

When an attribute has been created, it will automatically be available for searching. This means that in the **Local Search** ( ), you can select the attribute in the list of search criteria (see figure 10.14).
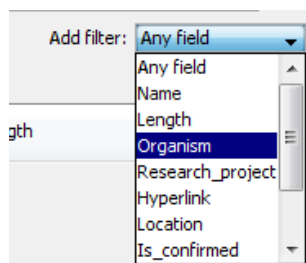
Figure 10.14: *The attributes from figure 10.11 are now listed in the search filter.*

It will also be available in the **Quick Search** below the **Navigation Area** (press Shift+F1 and it will be listed - see figure 10.15).
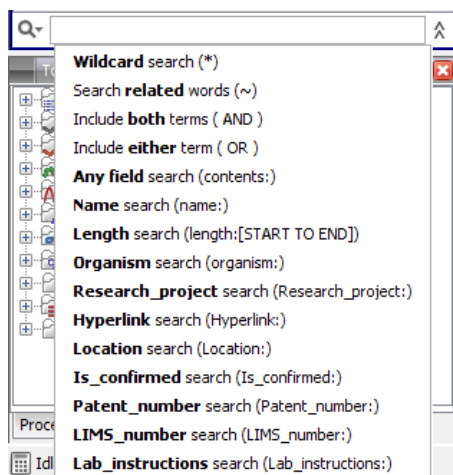
Figure 10.15: *The attributes from figure 10.11 are now available in the Quick Search as well.*

10.4 Local search

There are two ways of doing text-based searches of your data, as described in this section:

- **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.

10.4.1 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.

- **Custom attributes.** Read more in section 10.2

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 19.1.3) instead.

10.4.2   Quick search

At the bottom of the **Navigation Area** there is a text field as shown in figure 10.16).
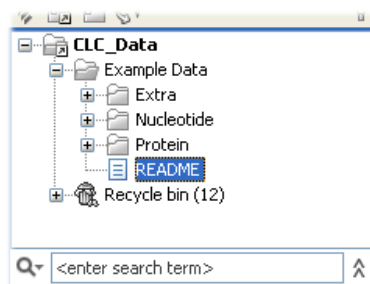
Figure 10.16: *Search simply by typing in the text field and press Enter.*

To search, simply enter a text to search for and press **Enter**.

Quick search results

To show the results, the search pane is expanded as shown in figure 10.17).

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 10.18).

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**.

In the preferences (see Chapter 11), you can specify the number of hits to be shown.

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 10.19.
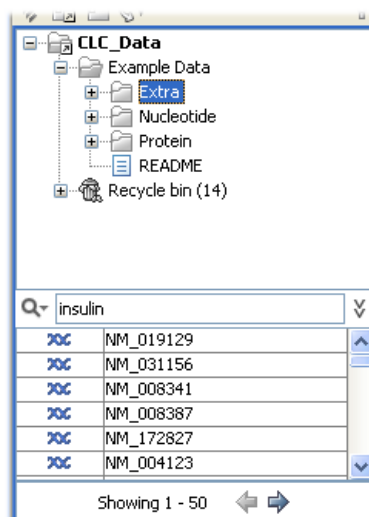
Figure 10.17: *Search results.*
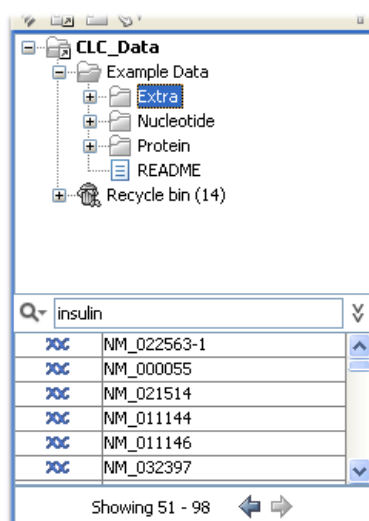
Figure 10.18: *Page two of the search results.*
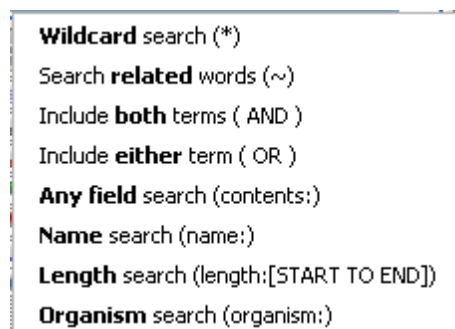
Figure 10.19: *Guides to help create advanced search expressions.*

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.

- **Do not include term (NOT)** If you write a term after not, then elements with these terms will not be returned.

- **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 17.4).

- **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".

Note! If you have added attributes (see section 10.2), these will also appear on the list when pressing **Shift+F1**.

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.
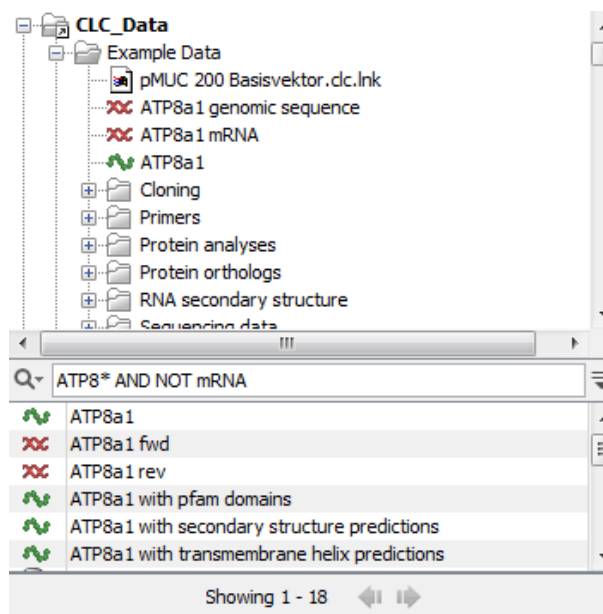
Figure 10.20: *An example of searching for elements with the name, description and organsim information that includes "ATP8" but do not include the term "mRNA".*

Search for data locations

The search function can also be used to search for a specific URL. This can be useful if you work on a server and wish to share a data location with another user. A simple example is shown in figure 10.21. Right click on the object name in the **Navigation Area** (in this case ATP8a1 genomic sequence) and select "Copy". When you use the paste function in a destination outside the Workbench (e.g. in a text editor or in an email), the data location will become visible. The URL can now be used in the search field in the Workbench to locate the object.
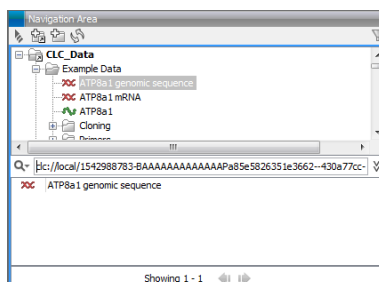
Figure 10.21: *The search field can also be used to search for data locations.*

Quick search history

You can access the 10 most recent searches by clicking the icon ($Q_{\blacktriangledown}$) next to the search field (see figure 10.22).
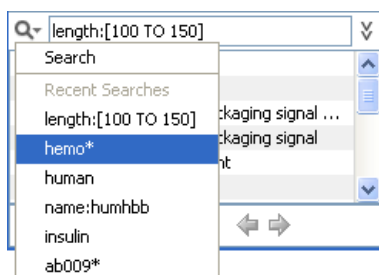
Figure 10.22: *Recent searches.*

Clicking one of the recent searches will conduct the search again.

10.4.3 Advanced search

As a supplement to the **Quick search** described in the previous section you can use the more advanced search:

>   **Edit | Local Search ( )**

> or **Ctrl + Shift + F (⌘ + Shift + F on Mac)**

This will open the search view as shown in figure 10.23

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 10.1.1.

Furthermore, you can specify what kind of elements should be searched:
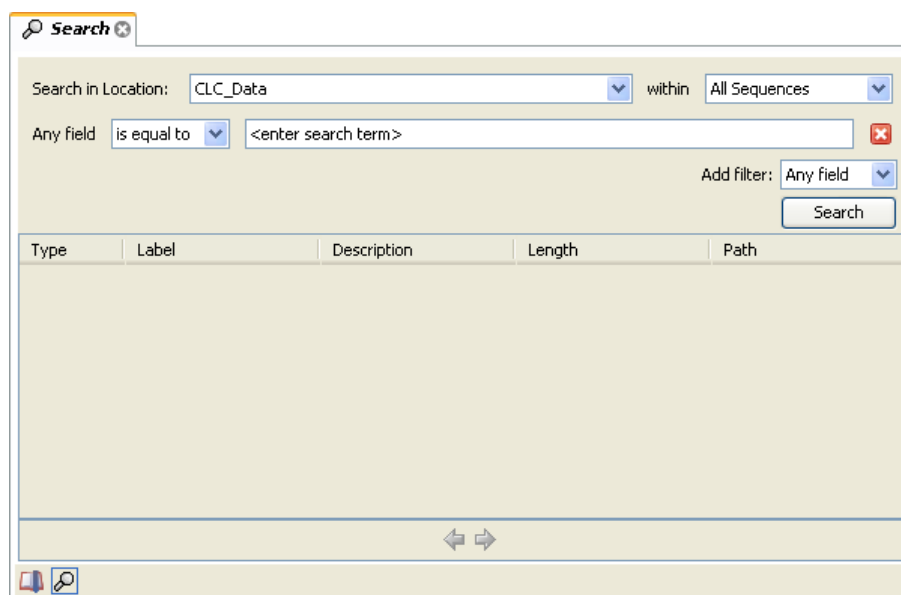
Figure 10.23: *Advanced search.*

- 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 10.4.2 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 17.4).

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 10.24.

Figure 10.24: *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.

10.4.4 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 Main 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 9.3.1.

Chapter 11

User preferences and settings

Contents

The first three sections in this chapter deal with the general preferences that can be set for *CLC Main 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 11.1:

> **Edit | Preferences (⚙)**

> or **Ctrl + K (⌘ + ; on Mac)**

11.1 General preferences

The **General** preferences include:
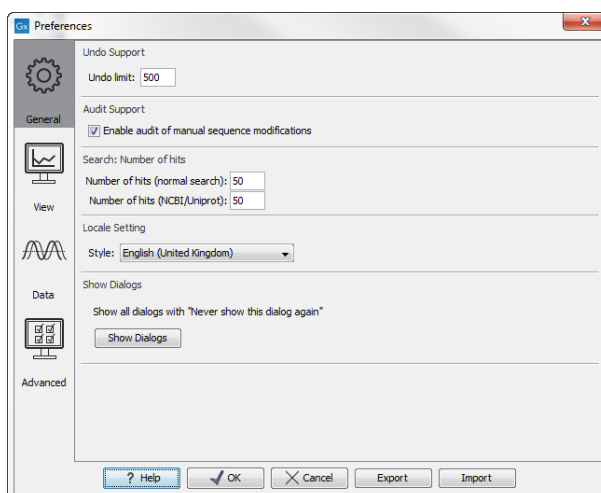
Figure 11.1: *Preferences include General preferences, View preferences, Data preferences, and Advanced settings.*

- **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 molecules, sequences, alignments or trees. See section 9.1.5 for more on this topic.

- **Audit Support.** If this option is checked, all manual editing of sequences will be marked with an annotation on the sequence (see figure 11.2). Placing the mouse on the annotation will reveal additional details about the change made to the sequence (see figure 11.3). Note that no matter whether **Audit Support** is checked or not, all changes are also recorded in the **History** ( ) (see section 14).

- **Number of hits.** The number of hits shown in *CLC Main 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.

Figure 11.2: *Annotations added when the sequence is edited.*

Figure 11.3: *Details of the editing.*

11.2   Default view preferences

There are six groups of default **View** settings:

1. **Toolbar**

2. **Show Side Panel**

3. **New View**

4. **Sequence Representation**

5. **User Defined View Settings**

6. **Molecule Project 3D Editor**

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 **Show Side Panel** setting allows you to choose whether to display the side panel.

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 **Sequence Representation** 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 11.6 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 11.4).

In this example, the **CLC Standard Settings** is chosen as default.

The **Molecule Project 3D Editor** gives you the option to turn off the modern OpenGL rendering for **Molecule Projects** (see section 21.6).
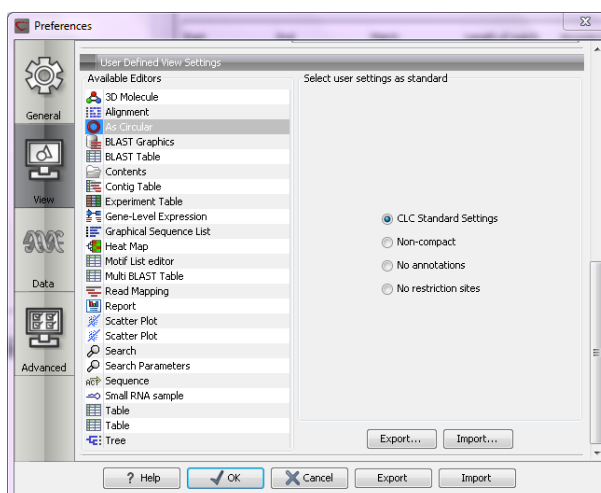
Figure 11.4: *Selecting the default view setting.*

11.2.1   Number formatting in tables

In the preferences, you can specify how the numbers should be formatted in tables (see figure 11.5).
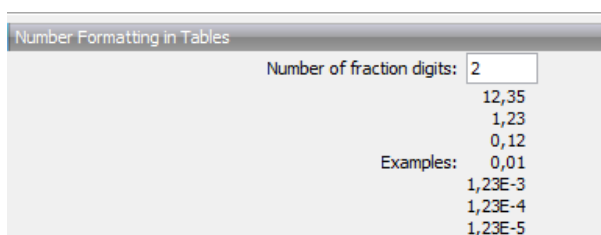
Figure 11.5: *Number formatting of tables.*

The examples below the text field are updated when you change the value so that you can see the effect. After you have changed the preference, you have to re-open your tables to see the effect.

11.2.2   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.

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 11.5).

A dialog will be shown (see figure 11.6) that allows you to select which of the settings you wish to export.

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
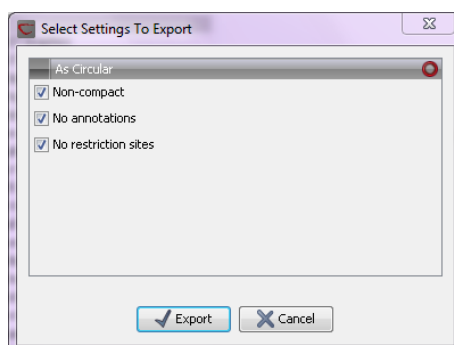
Figure 11.6: *Exporting all settings for circular views.*

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 11.5).

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 11.7).

Figure 11.7: *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.*

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:

- Import and export of **bioinformatics data** such as sequences, alignments etc. (described in section 13.1).

- **Graphics** export of the views which creates image files in various formats (described in section 13.3).

- 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.

11.3 Data preferences

The data preferences contain preferences related to interpretation of data, e.g. linker sequences:

- Predefined primer additions for Gateway cloning (see section 28.2.1).

11.4   Advanced preferences

The **Advanced** settings include the possibility to set up a proxy server. This is described in section 1.8.

11.4.1   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.

The default data location for CLC Workbenches is, by default, a folder called CLC_Data in a user's home area.

This can be changed to a different location for a particular user of the Workbench by going to

>   **Edit** | **Preferences**

and then choosing the **Advanced** tab. This holds a section called **Default Data Location** and here you can choose a default from a drop down list of data locations you have already added.

Note! The default location cannot be removed. You have to select another location as default first.

If the data area you want as your default is not already available in your Workbench, you need to first add it as a new data location (see section 10.1.1).

11.4.2   NCBI BLAST

URL to use for BLAST

It is possible to specify an alternate server URL to use for BLAST searches. The standard URL for the BLAST server at NCBI is: http://blast.ncbi.nlm.nih.gov/Blast.cgi.

Note! Be careful to specify a valid URL, otherwise BLAST will not work.

11.5   Export/import of preferences

The user preferences of the *CLC Main 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 11.2.2.

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**

11.5.1   The different options for export and import

To avoid confusion of the different import and export options, you can find an overview here:

- Import and export of **bioinformatics data** such as molecules, sequences, alignments etc. (described in section 13.1).

- **Graphics** export of the views that create image files in various formats (described in section 13.3).

- 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.

11.6   View settings for the Side Panel

The **Side Panel** is shown to the right of all views that are opened in *CLC Main Workbench* and is described in further detail in section 9.1.8.

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 at the bottom of the **Side Panel** (see figure 11.8).
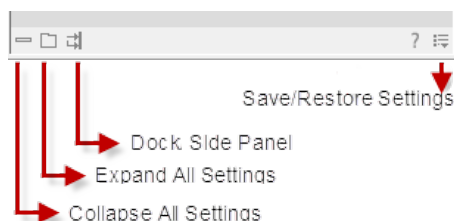
Figure 11.8: *At the bottom of the Side Panel you save the view settings*

11.6.1   Saving, removing and applying saved settings

To save and apply the saved settings, click ( ) seen in figure 11.8. This opens a menu where the following options are available (figure 11.9):

- **Save ... Settings.** ( ) The settings can be saved in two different ways. When you select either way of saving settings a dialog will open (see figure 11.10) where you can enter a name for your settings.

    - **For ... View in General** ( ) Will save the currently used settings with all elements of the same type as the one used for adjusting the settings. E.g. if you have selected to save settings "For Track View in General" the settings will be applied each time you open an element of the same type, which in this case means each time one of the
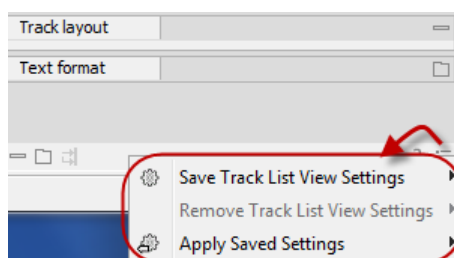
Figure 11.9: *When you have adjusted the side panel settings and would like to save these, this can be done with the "Save ... Settings" function, where "..." is the element you are working on - e.g. "Track List View", "Sequence View", "Table View", "Alignment View" etc. Saved settings can be deleted again with "Remove ... Settings" and can be applied to other elements with "Apply Saved Settings".*

saved tracks are opened from the **Navigation Area**. These "general" settings are user specific and will not be saved with or exported with the element.

– **On This Only** ( ) Settings can be saved with the specific element that you are working on in the View area and will not affect any other elements (neither in the View Area or in the **Navigation Area**). E.g. for a track you would get the option to save settings "On This Track Only". The settings are saved with only this element (and will be exported with the element if you later select to export the element to another destination).
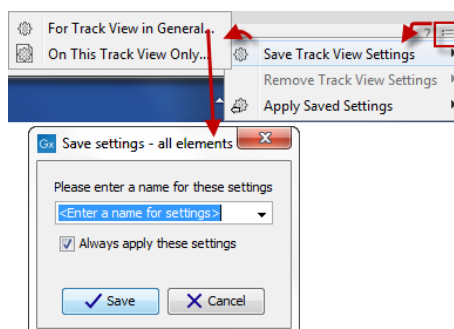
Figure 11.10: *The save settings dialog.Two options exist for saving settings. Click on the relevant option to open the dialog shown at the bottom of the figure.*

• **Remove ... Settings.** ( ) Gives you the option to remove settings specifically for the element that you are working on in the View Area, or on all elements of the same type. When you have selected the relevant option, the dialog shown in figure 11.11 opens and allows you to select which of the saved settings to remove.

– **From ... View in General** ( ) Will remove the currently used settings on all elements of the same type as the one used for adjusting the settings. E.g. if you have selected to remove settings from all alignments using "From Alignment View in General", all alignments in your **Navigation Area** will be opened with the standard settings in stead.

– **From This ... Only** ( ) When you select this option, the selected settings will only be removed from the particular element that you are working on in the View area and will not affect any other elements (neither in the View Area or in the **Navigation Area**). The settings for this particular element will be replaced with the CLC standard settings ( ).
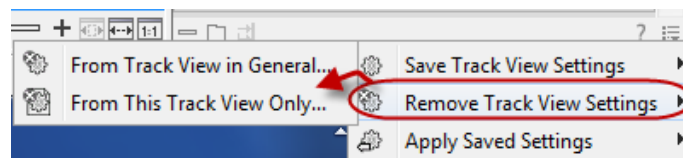
Figure 11.11: *The remove settings dialog for a track.*

- **Apply Saved Settings.**   (⚙) This is a submenu containing the settings that you have previously saved (figure 11.12). 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.   (⚙)
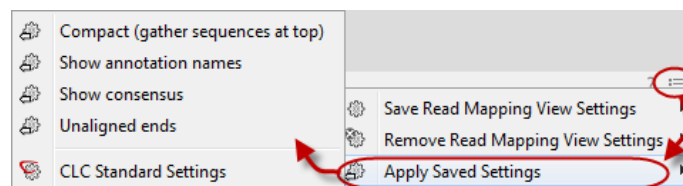
Figure 11.12: *Applying saved settings.*

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 11.2.2).

Chapter 12

Printing

Contents

CLC Main Workbench offers different choices of printing the result of your work.

This chapter deals with printing directly from *CLC Main Workbench*. Another option for using the graphical output of your work, is to export graphics (see chapter 13.3) 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 Main 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 12.1).

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.
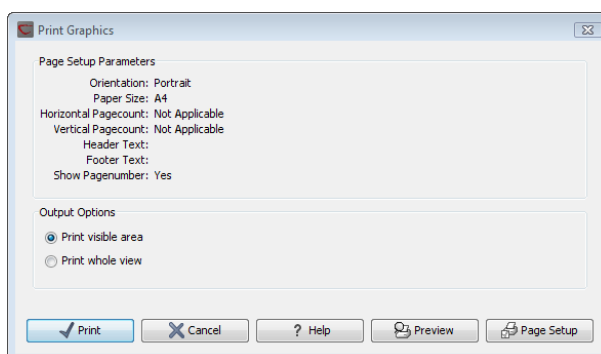
Figure 12.1: *The Print dialog.*

12.1 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 12.2 is a view of a circular sequence which is zoomed in so that you can only see a part of it.
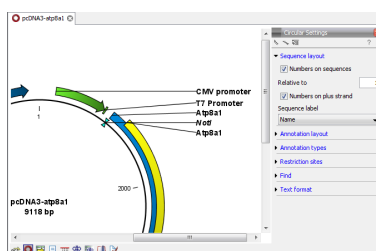
Figure 12.2: *A circular sequence as it looks on the screen.*

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 12.2 and choosing **Print visible area** can be seen in figure 12.3.
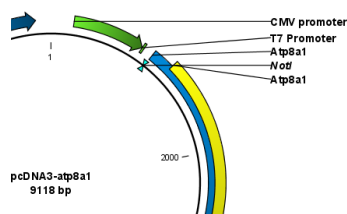
Figure 12.3: *A print of the sequence selecting Print visible area.*

On the other hand, if you select **Print whole view**, you will get a result that looks like figure 12.4. This means that you also print the part of the sequence which is not visible when you have zoomed in.
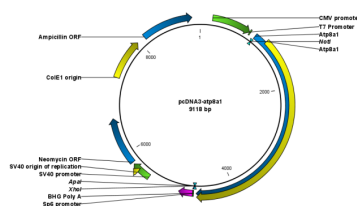
Figure 12.4: *A print of the sequence selecting Print whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*

12.2   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 12.5
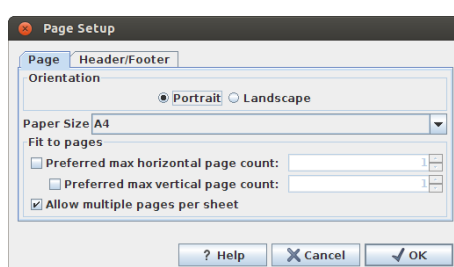
Figure 12.5: *Page Setup.*

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 12.6 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 12.6: *An example where Fit to pages horizontally is set to 2, and Fit to pages vertically is set to 3.*

12.2.1 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.

12.3 Print preview

The preview is shown in figure 12.7.
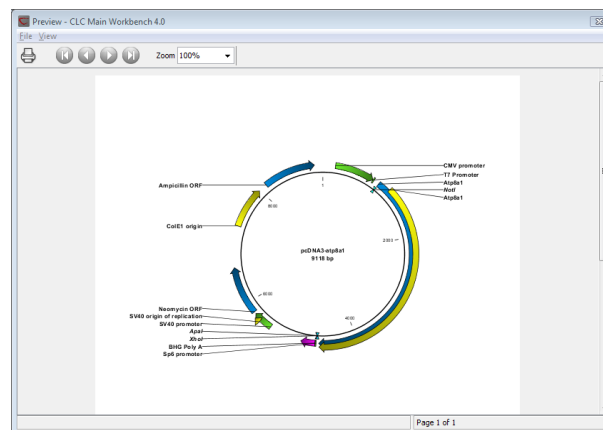
Figure 12.7: *Print preview.*

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 13

Import/export of data and graphics

Contents

CLC Main Workbench handles a large number of different data formats. In order to work with data in the Workbench, it has to be imported (![icon]). Data types that are not recognized by the Workbench are imported as "external files" which means that when you open these, they will open in the default application for that file type on your computer (e.g. Word documents will open in Word).

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 to export graphics.

13.1   Standard import

CLC Main Workbench has support for a wide range of bioinformatic data such as molecules, sequences, alignments etc. See a full list of the data formats in section I.1.

These data can be imported through the Import dialog, using drag/drop or copy/paste as explained below.

13.1.1   Import using the import dialog

To start the import using the import dialog:

click Import (📥) in the Toolbar

This will show a dialog similar to figure 13.1. You can change which kind of file types that should be shown by selecting a file format in the **Files of type** box.
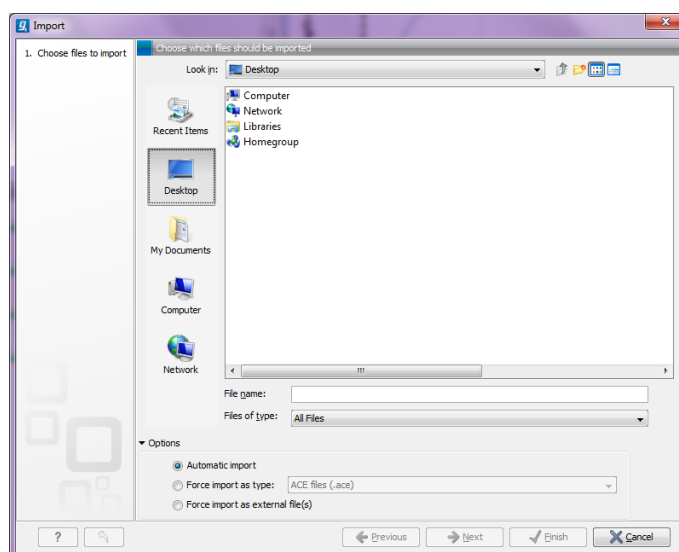
Figure 13.1: *The import dialog.*

Next, select one or more files or folders to import and click **Next**.

This allows you to select a place for saving the result files.

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 13.1), there are three import options:

Automatic import This will import the file and *CLC Main 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 Main 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.

13.1.2 Import using drag and drop

It is also possible to drag a file from e.g. the desktop into the **Navigation Area** of *CLC Main 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.

13.1.3 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 Main 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 Main Workbench*.

If the sequence is not formatted, i.e. if you just have a text like this: "ATGACGAATAGGAGTTC-TAGCTA" you can also paste this into the **Navigation Area**.

Note! Make sure you copy all the relevant text - otherwise *CLC Main Workbench* might not be able to interpret the text.

13.1.4 External files

In order to help you organize your research projects, *CLC Main 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 Main 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 13.1). Bioinformatics files not recognized by *CLC Main Workbench* are also treated as external files.

13.1.5 Import Vector NTI data
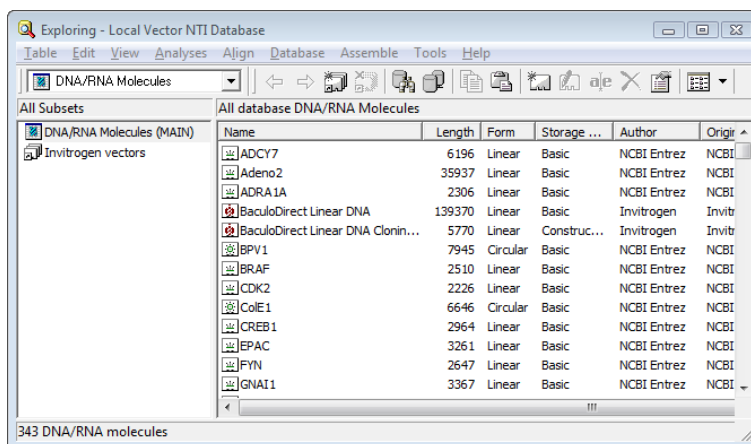
There are several ways of importing your Vector NTI data into the CLC Workbench. The best way to go depends on how your data is currently stored in Vector NTI:

- 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 13.2), you can import all the data in one step, or you can import selected parts of it.

Figure 13.2: *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 13.3):

File | Import Vector NTI Database
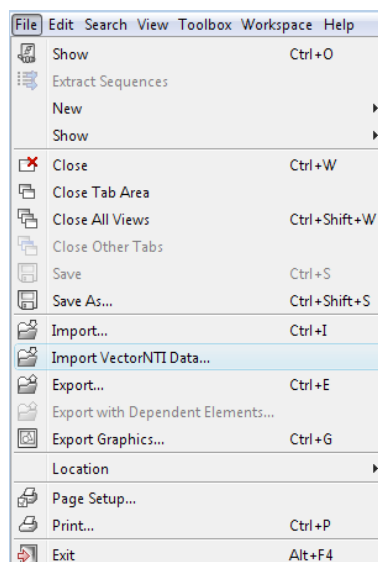
Figure 13.3: *Import the whole Vector NTI Database.*

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 13.4.
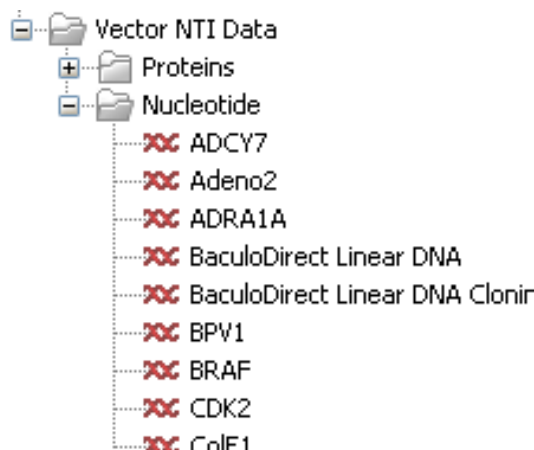
Figure 13.4: *The Vector NTI Data folder containing all imported sequences of the Vector NTI Database.*

If something goes wrong during the import process, please report the problem to support@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 13.5.
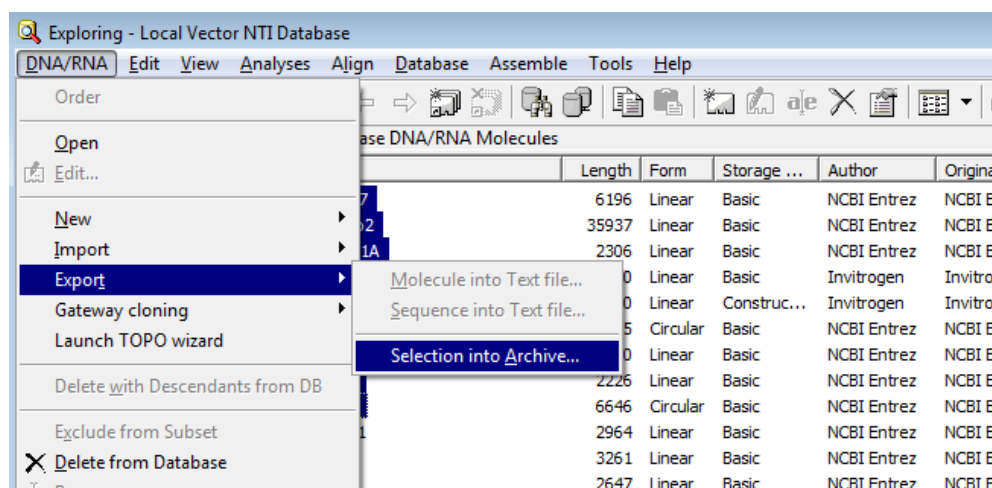
Figure 13.5: *Select the relevant files and export them as an archive through the File menu.*

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 13.6).
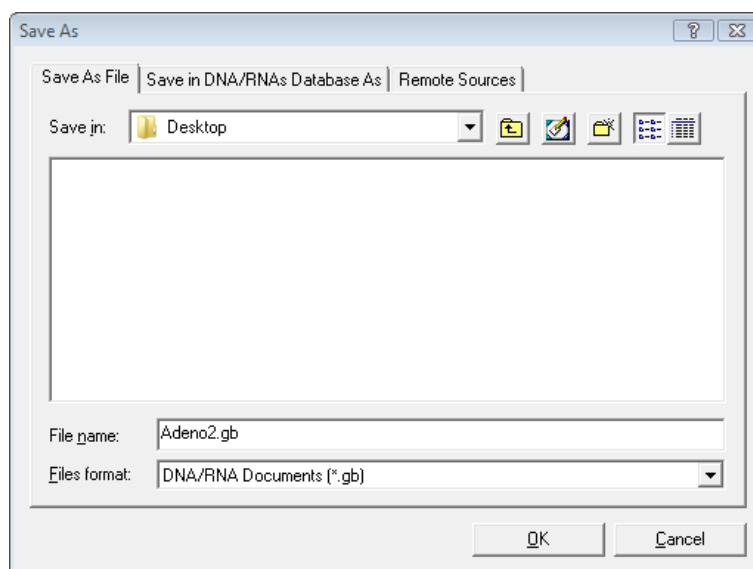
Figure 13.6: *Saving a sequence as a file in Vector NTI.*

This will give you file with a .gb extension. This file can be easily imported into the CLC Workbench:

> **Import (⤵) | select the file | Select**

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 plugin which is pre-installed in the Workbench. It can be uninstalled and updated using the plugin manager (see section 1.7).

13.2   Data export

The exporter can be used to:

- Export bioinformatic data in most of the formats that can be imported. There are a few exceptions (see section I.1).

- Export one or more data elements at a time to a given format. When multiple data elements are selected, each is written out to an individual file, unless compression is turned on.

The standard export functionality can be launched using the Export button on the toolbar, or by going to the menu:

> **File | Export (⤴)**

An additional export tool is available from under the File menu:

> **File | Export with Dependent Elements**

This tool is described further in section 13.2.2.

The general steps when configuring a standard export job are:

- (Optional) Select the data to export in the **Navigation Area**.

- Start up the exporter tool via the Export button in the toolbar or using the **Export** option under the File menu.

- Select the format the data should be exported to.

- Select the data to export, or confirm the data to export if it was already selected via the **Navigation Area**.

- Configure the parameters. This includes compression and naming of the output files, along with other format-specific settings where relevant.

- Select where the data should be exported to.

- Click on the button labeled **Finish**.

Selecting data for export - part I. You can select the data elements to export **before** you run the export tool **or after** the format to export to has been selected. If you are not certain which formats are supported for the data being exported, we recommend selecting the data in the **Navigation Area** before launching the export tool.

Selecting a format to export to. When data is pre-selected in the **Navigation Area** before launching the export tool you will see a column in the export interface called **Supported formats**. Formats that the selected data elements can be exported to are indicated by a "Yes" in this column. Supported formats will appear at the top of the list of formats. See figure 13.7.

Name	Description	Extension	Supported format
Fasta	Export sequences and sequence lists in fasta format	[fa, fsa, fasta]	Yes
Fastq	Export sequences and sequence lists in fastq format	[fastq]	Yes
GFF	Export sequence annotations in General Feature Format	[gff]	Yes
GenBank	Export sequences and sequence lists in GenBank format	[gbk, gb, gp]	Yes
General Transfer Format	Export Gene, CDS and mRNA combined in Gene Transfer F...	[GTF]	Yes
HTML	Export tables and tabular information in HTML	[html]	Yes
History PDF	Export the history of an element in Portable Document For...	[pdf]	Yes
Nexus	Export phylogenetic trees in Nexus format	[nxs, nexus]	Yes
PIR	Export sequences and sequence lists in PIR format	[pir]	Yes
Sequence CSV	Export sequences or sequence lists as Comma Separated ...	[csv]	Yes
Tab delimited text	Export tables or tabular information as Tab Delimited Text	[txt]	Yes
Table CSV	Export tables in CSV format	[csv]	Yes
Zip export	Export files and folder structure in CLC format to a zip file	[zip]	Yes
ACE	Export sequencing reads mapped to a reference or de nov...	[ace]	No
BAM	Export mapped reads in BAM format	[bam]	No
ClustalW	Export alignments in ClustalW format	[aln]	No

Figure 13.7: *The Select exporter dialog where sequence lists were pre-selected in the Navigation Area before launching the export tool. Here, the formats sequence lists can be exported to are listed at the top, with a Yes in the Selected formats column. Other formats are found below, with No in this column.*

Formats that cannot be used for export of the selected data have a "No" listed in the **Supported formats** column. If you have selected multiple data elements of different types, then formats which can be used for some of the selected data elements but not all of them are indicated by the text "For some elements" in this column.

Please note that the information in the **Supported formats** column only refers to the data already selected in the **Navigation Area**. If you are going to choose your data later in the export process, then the information in this column will not be pertinent.

Only one export format is available if you select a folder to be exported. This is described in more detail in section 13.2.1.

Finding a particular format in the list. You can quickly find a particular format by using the text box at the top of the exporter window as shown in figure 13.8, where formats that include the term VCF are searched for. This search term will remain in place the next time the Export tool is launched. Just delete the text from the search box if you no longer wish only the formats with that term to be listed.

When the desired export format has been identified, click on the button labeled **Open**.

Selecting data for export - part II. A dialog appears, with a name reflecting the format you have chosen. For example if the "Variant Call Format" (VCF format) was selected, the window is labeled "Export VCF".

If you are logged into a CLC Server, you will be asked whether to run the export job using the Workbench or the Server. After this, you are provided with the opportunity to select or de-select data to be exported.

In figure 13.9 we show the selection of a variant track for export to VCF format.
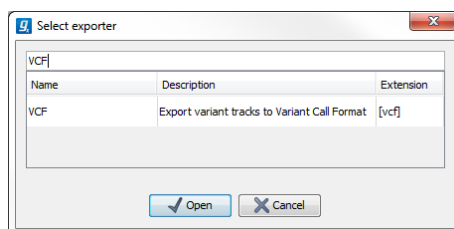
Figure 13.8: *The text field has been used to search for VCF format in the Select exporter dialog.*
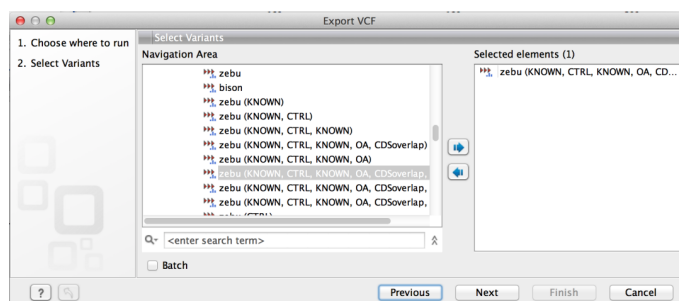
Figure 13.9: *The Select exporter dialog. Select the data element(s) to export.*

The parameters under **Basic export parameters** and **File name** are offered when exporting to any format. There may be additional parameters for particular export formats. This is illustrated here with the VCF exporter, where a reference sequence track must be selected. See figure 13.10.

Compression options. Within the **Basic export parameters** section, you can choose to compress the exported files. The options are no compression (None), gzip or zip format. Choosing zip format results in all data files being compressed into a single file. Choosing gzip compresses the exported file for each data element individually.

Choosing the exported file name(s) The default setting for the **File name** is to use the original data element name as the basename and the export format as the suffix.

When exporting just one data element, or exporting to a zip file, the desired filename could just be typed in the Custom file name box.
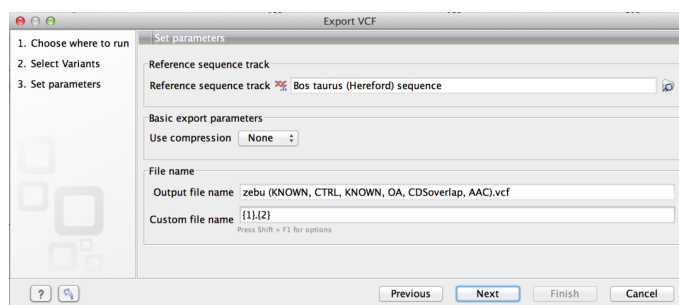
Figure 13.10: *Set the export parameters. When exporting in VCF format, a reference sequence track must be selected.*

When working with the export of multiple files, using some combination of the terms shown by default in this field and in figure 13.11 are recommended. Clicking in the **Custome file name** field with them mouse and then simultaneously pressing the Shift + F1 keys bring up a list of the available terms that can be included in this field.

As you add or remove text and terms in the **Custome file name** field, the text in the **Output file name** field will change so you can see what the result of your naming choice will be for your data. When working with multiple files, only the name of the first one is shown. Just move the mouse cursor over the name shown in the **Output file name** field to show a listing of the all the filenames.
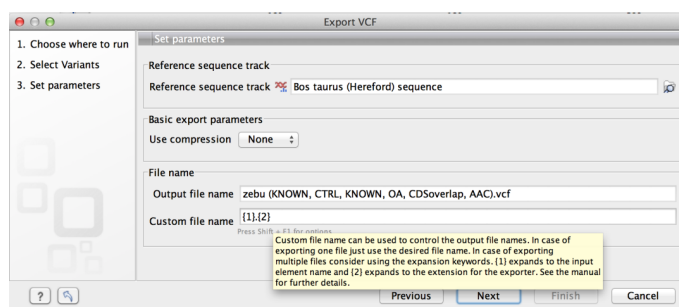
Figure 13.11: *Use the custom file name pattern text field to make custom names.*

The last step is to specify the exported data should be saved (figure 13.12).
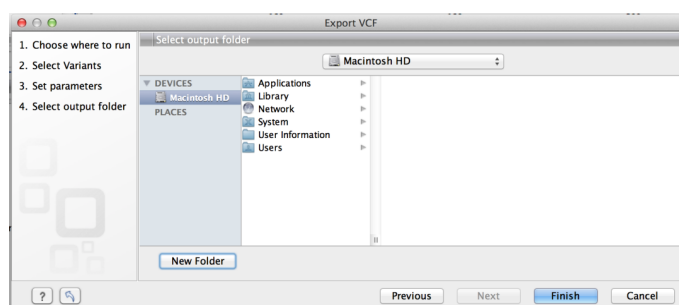
Figure 13.12: *Select where to save the exported data.*

A note about decimals and Locale settings. When exporting to CSV and tab delimited files, decimal numbers are formatted according to the Locale setting of the Workbench (see section 11.1). 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.

13.2.1   Export of folders and multiple elements in CLC format

In the list of export formats presented is one called zip format. Choosing this format means that you wish to export the selected data element(s) or folders to a single, compressed CLC format file. This is useful in cases where you wish to exchange data between workbenches or as part of a simple backup procedure.

A zip file generated this way can be imported directly into a CLC Workbench using the Standard Import tool and leaving the import type as Automatic.

Note! When exporting multiple files, the names will be listed in the "Output file name" text field with only the first file name being visible and the rest being substituted by "...", but will appear in a tool tip if you hover the mouse over that field (figure 13.13).
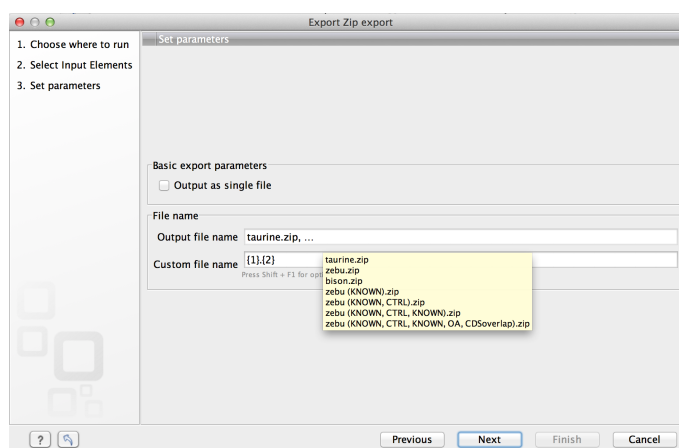
Figure 13.13: *The output file names are listed in the "Output file name" text field.*

13.2.2   Export of dependent elements

Sometimes it can be useful to export the results of an analysis and its dependent elements. That is, the results along with the data that was used in the analysis. For example, one might wish to export an alignment along with all the sequences that were used in generating that alignment.

To export a data element with its dependent elements:

- Select the parent data element (like an alignment) in the **Navigation Area**.

- Start up the exporter tool by going to F**File | Export with Dependent Elements**.

- Edit the output name if desired and select where the resulting zip format file should be exported to.

The file you export contains compressed CLC format files containing the data element you chose and all its dependent data elements.

A zip file created this way can be imported directly into a CLC workbench by going to

> **File | Import | Standard Import**

In this case, the import type can be left as Automatic.

13.2.3   Export history

Each data element in the Workbench has a history. The history information includes things like the date and time data was imported or an analysis was run, the parameters and values set, and where the data came from. For example, in the case of an alignment, one would see the sequence data used for that alignment listed. You can view this information for each data element by clicking on the Show History view ( ) at the bottom of the viewing area when a data element is open in the Workbench.

This history information can be exported to a pdf document. To do this:

- (Optional, but preferred) Select the data element (like an alignment) in the **Navigation Area**.

- Start up the exporter tool via the Export button in the toolbar or using the **Export** option under the File menu.

- Select the **History PDF** as the format to export to. See figure 13.14.

- Select the data to export, or confirm the data to export if it was already selected via the **Navigation Area**.

- Edit any parameters of interest, such as the Page Setup details, the output filename(s) and whether or not compression should be applied. See figure 13.15.

- Select where the data should be exported to.

- Click on the button labeled **Finish**.
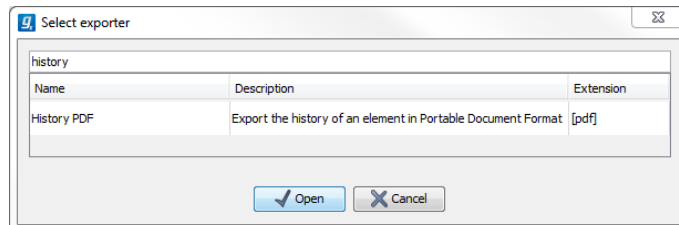
Figure 13.14: *Select "History PDF" for exporting the history of an element.*
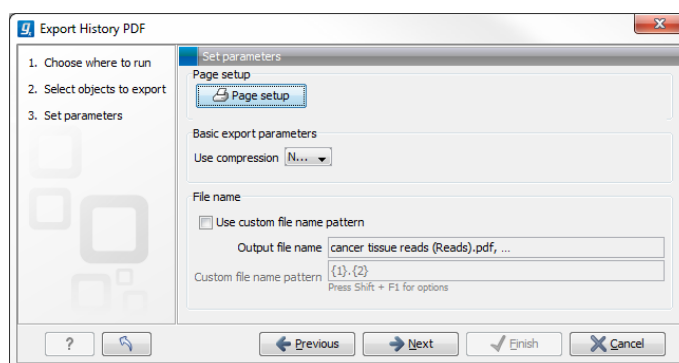
Figure 13.15: *When exporting the history in PDF, it is possible to adjust the page setup.*

13.2.4   The CLC format

The *CLC Main Workbench* stores bioinformatic data in CLC format. The CLC format contains data, as well as information about that data like history information and comments you may have added.

A given data element in the Workbench can contain different types of data. This is reflected when exporting data, as the choice of different export formats can lead to the extraction of some parts of that data object rather than others. The part of the data exported reflects the type of data a given format can support. As a simple example, if you export the results of an alignment to Annotation CSV format, you will get just the annotation information. If you exported to Fasta alignment format, you would get the aligned sequences in fasta format, but no annotations.

The CLC format holds all the information for a given data object. Thus if you plan to share the data with colleagues who also have a CLC Workbench or you are communicating with the CLC Support team and you wish to share the data from within the Workbench, exporting to CLC format is usually the best choice as all information associated with that data object in your Workbench will then be available to the other person who imports that data.

If you are planning to share your data with someone who does not have access to a CLC Workbench, then you will wish to export to another data format. Specifically, one they can use with the software they are working with.

13.2.5   Backing up data from the CLC Workbench

Regular backups of your data are advisable.

The data stored in your CLC Workbench is in the areas defined as CLC Data Locations. Whole data locations can be backed up directly (option 1) or, for smaller amounts of data, you could export the selected data elements to a zip file (option 2).

Option 1: Backing up each CLC Data Location

The easiest way for most people to find out where their data is stored is to put the mouse cursor over the top level directories, that is, the ones that have an icon like ( ), in the **Navigation Area** of the Workbench. This brings up a tool tip with the system location for that data location.

To back up all your CLC data, please ensure that all your CLC Data Locations are backed up.

Here, if you needed to recover the data later, you could put add the data folder from backup as a data location in your Workbench. If the original data location is not present, then the data should be usable directly. If the original data location is still present, the Workbench will re-index the (new) data location. For large volumes of data, re-indexing can take some time.

Information about your data locations can also be found in an xml file called model_settings_300.xml This file is located in the settings folder in the user home area. Further details about this file and how it pertains to data locations in the Workbench can be found in the Deployment Manual:

http://www.clcsupport.com/workbenchdeployment/current/index.php?manual=
Changing_default_location.html

Option 2: Export a folder of data or individual data elements to a CLC zip file

This option is for backing up smaller amounts of data, for example, certain results files or a whole data location, where that location contains smaller amounts of data. For data that takes

up many gigabases of space, this method can be used, but it can be very demanding on space, as well as time.

Select the data items, including any folders, in the Navigation area of your Workbench and choose to export by going to:

File | **Export ( )**

and choosing ZIP format.

The zip file created will contain all the data you selected. You can later re-import the zip file into the Workbench by going to:

File | **Import ( )**

The only data files associated with the *CLC Main Workbench* not within a specified data location are BLAST databases. It is unusual to back up BLAST databases as they are usually updated relatively frequently and in many cases can be easily re-created from the original files or re-downloaded from public resources. If you do wish to backup your BLAST database files, they can be found in the folders specified in the BLAST Database Manager, which is started by going to **Toolbox | BLAST | Manage BLAST databases**.

13.2.6 Export of workflow output

The output from a workflow can be exported by adding one or more workflow export elements (figure 13.16). Multiple elements can be selected by holding down the Ctrl key while clicking on the desired elements.
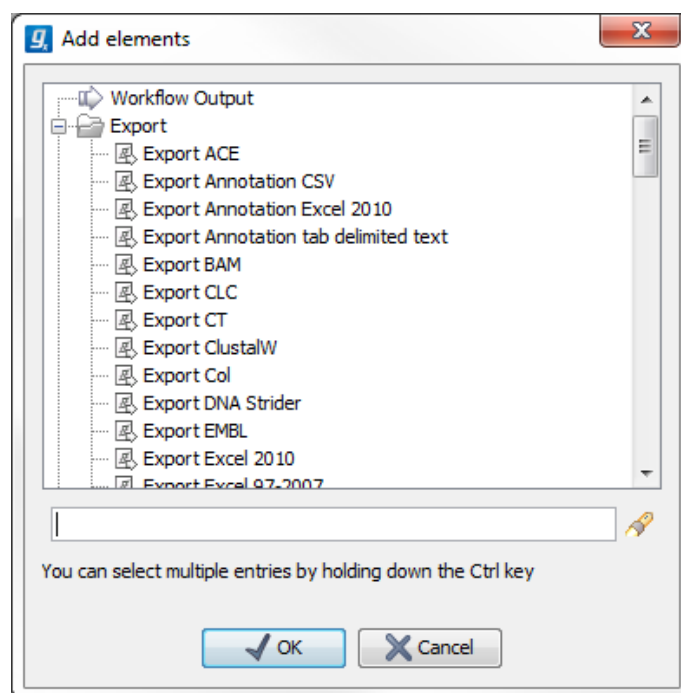
Figure 13.16: *Pressing "Add element" enables addition of workflow export elements.*

When the workflow has been created, you can set the export parameters and the location to export data to by double clicking on each export element or leave fields empty and unlocked if you wish users of the Workflow to enter this information when the Workflow is launched.
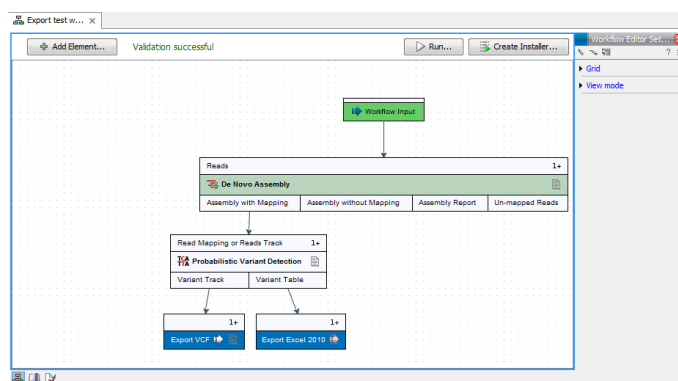
Figure 13.17: *A simple workflow with two export elements. The variant track will be exported in VCF format and the variant table in Excel format.*

13.3   Export graphics to files

CLC Main 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 Main 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 13.18.
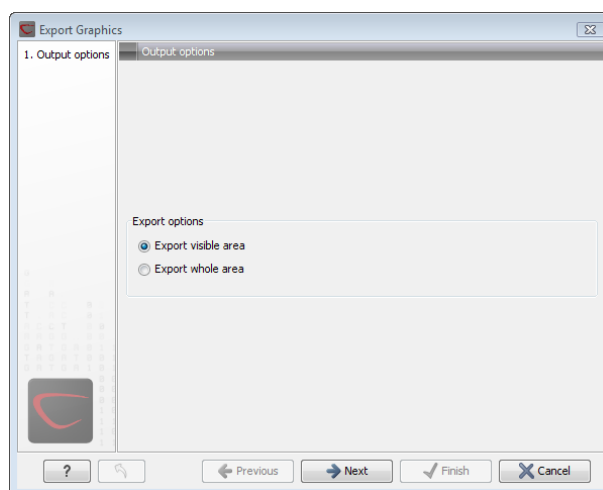
Figure 13.18: *Selecting to export whole view or to export only the visible area.*

13.3.1   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 13.19 is a view of a circular sequence which is zoomed in so that you can only see a part of it.
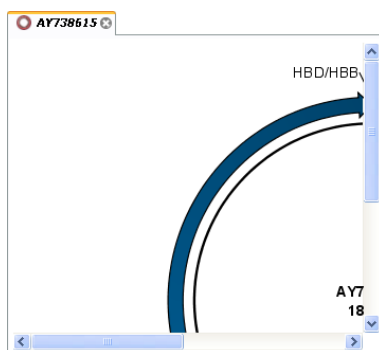
Figure 13.19: *A circular sequence as it looks on the screen.*

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 13.19 and choosing **Export visible area** can be seen in figure 13.20.
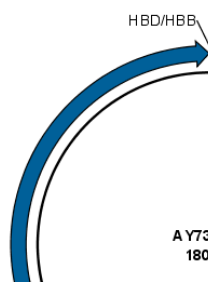
Figure 13.20: *The exported graphics file when selecting Export visible area.*

On the other hand, if you select **Export whole view**, you will get a result that looks like figure 13.21. 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.

13.3.2 Save location and file formats

In this step, you can choose name and save location for the graphics file (see figure 13.22).

CLC Main Workbench supports the following file formats for graphics export:
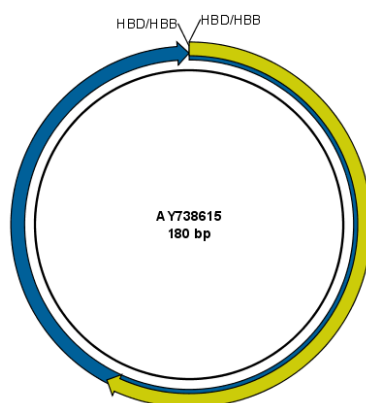
Figure 13.21: *The exported graphics file when selecting Export whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*
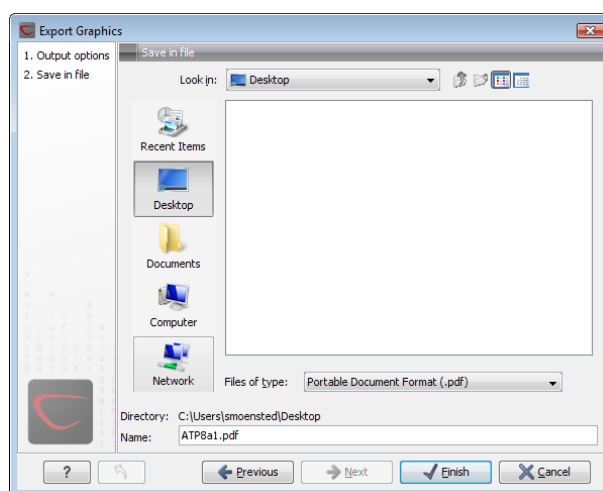
Figure 13.22: *Location and name for the graphics file.*

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 Main Workbench*. See section 13.1.4 for more about importing external files into *CLC Main Workbench*.

13.3.3   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 13.23.
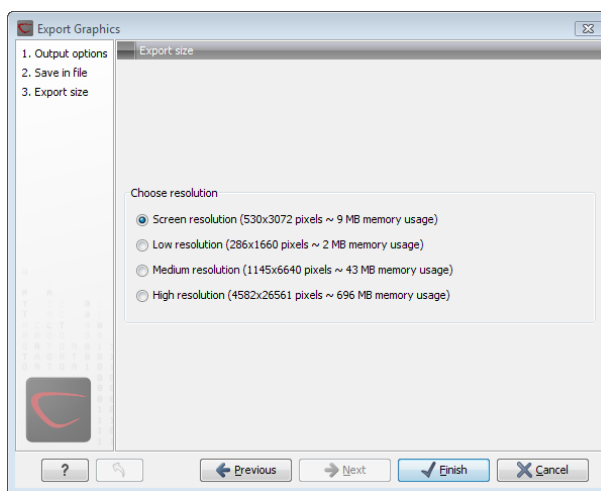
Figure 13.23: *Parameters for bitmap formats: size of the graphics file.*

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 13.24 (this is only the case if the graphics is using more than one page).
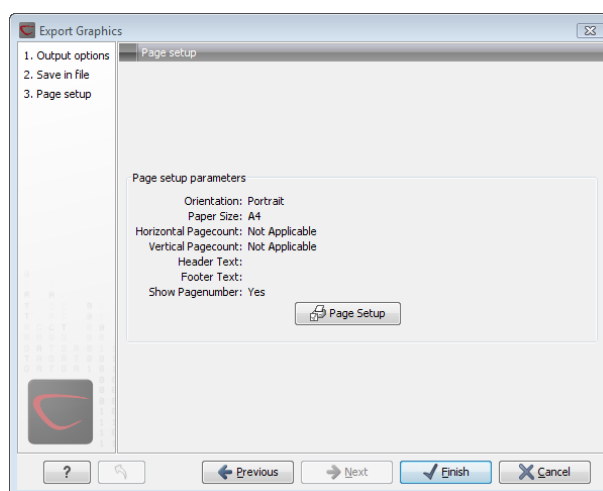
Figure 13.24: *Page setup parameters for vector formats.*

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 12.2.

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.

13.3.4 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.

13.4 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 13.25. This graph shows the coverage of reads of a read mapping (produced with *CLC Genomics Workbench*).

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

Figure 13.25: *A graph displayed along the mapped reads. Right-click the graph to export the data points to a file.*

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 13.26 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.
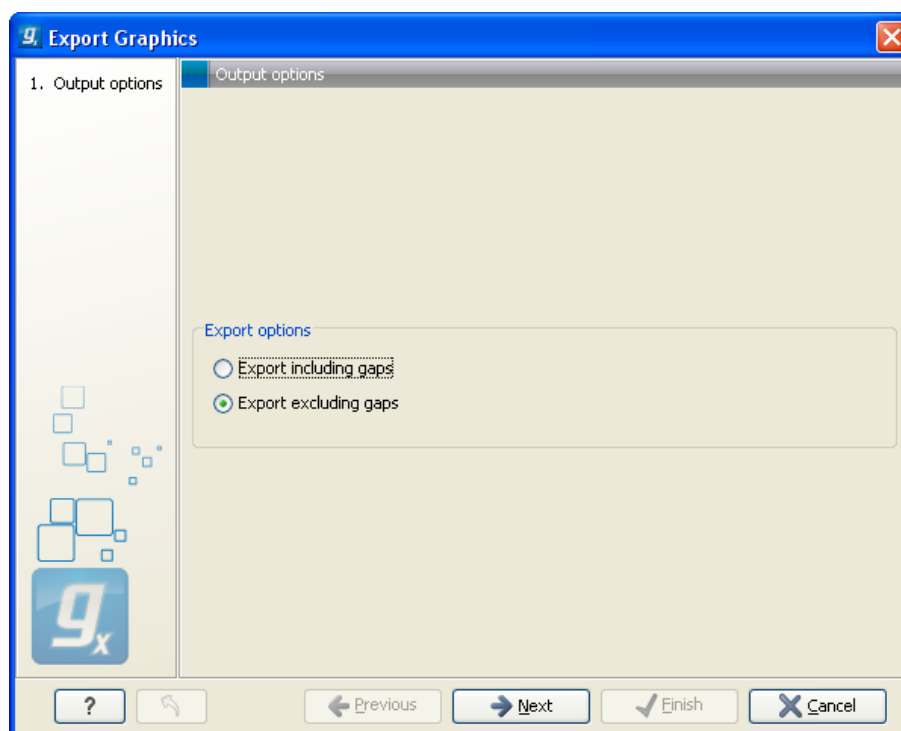
Figure 13.26: *Choosing to include data points with gaps*

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";
...
```
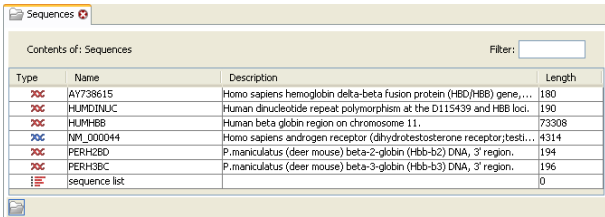
13.5   Copy/paste view output

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 Main 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 13.27.

Figure 13.27: *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 Main 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 14

History log

Contents

CLC Main 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 Main Workbench*.

14.1 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 Main 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 14.1.

When 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
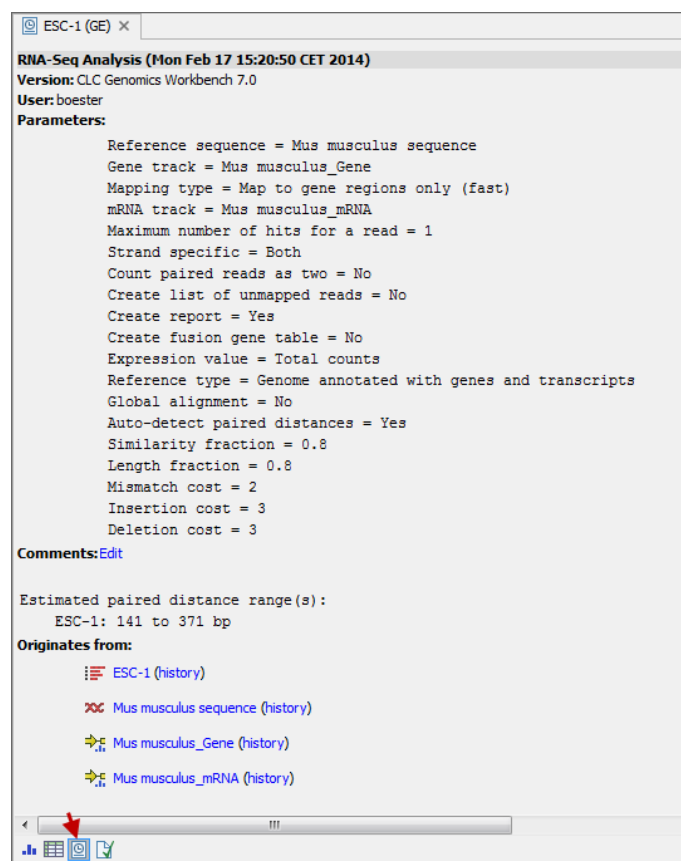
Figure 14.1: *An element's history.*

to your locale settings (see section 11.1).

- **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.

14.1.1   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 13.2.2.

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 15

Batching and result handling

Contents

15.1 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 available in *CLC Genomics Workbench* and supply five sequences as shown in figure 15.1, 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 the 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:
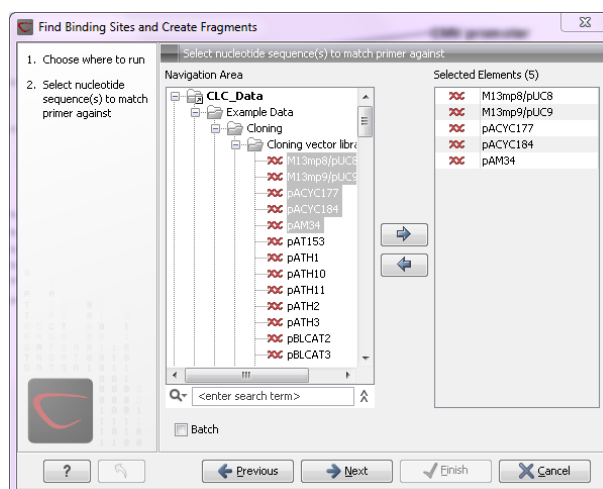
Figure 15.1: *Inputting five sequences to Find Binding Sites and Create Fragments.*

- 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 15.2.
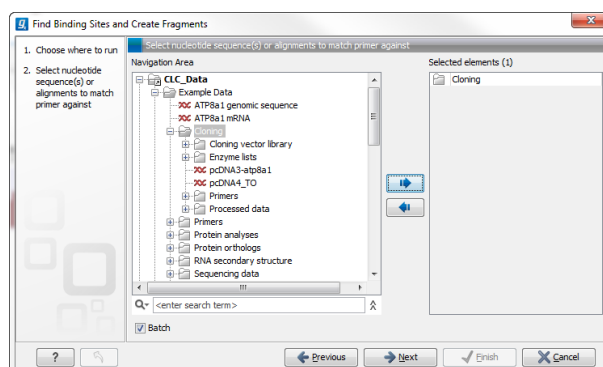
Figure 15.2: *The Cloning folder includes both folders and sequences.*

The `Cloning` folder that is found in the example data (see section 1.6.2) contains two sequences (✖) and four 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.

15.1.1   Batch overview

The batch overview lists the batch units to the left and the contents of the selected unit to the right (see figure 15.3).

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 four folders in the Cloning folder (see
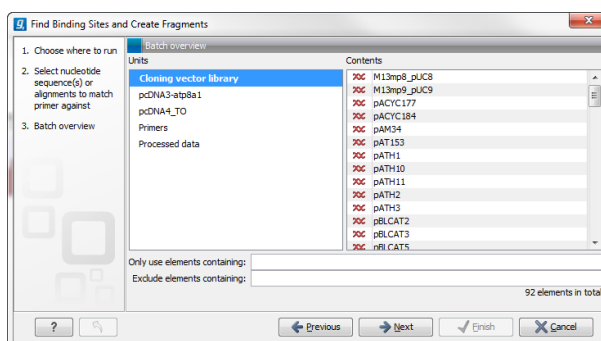
Figure 15.3: *Overview of the batch run.*

figure 15.2), and three 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).

15.1.2   Batch filtering and counting

At the bottom of the dialog shown in figure 15.3, the Workbench counts the number of files that will be run in total (92 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**.

15.1.3   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.

15.1.4 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, the default behavior is to place the result files in the same folder as the input files. In the example shown in figure 15.3, 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.

However, there is an option to save the results in a separate folder structure by checking **Into separate folders**. This will allow you to specify a new save destination, and the *CLC Main Workbench* will create a subfolder for each batch unit where the results are saved..

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.

15.2 How to handle results of analyses

This section will explain how results generated from tools in the Toolbox are handled by *CLC Main 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 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.

In this step, shown in figure 15.4, 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 15.5). 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.

15.2.1 Table outputs

Some analyses also generate a table with results, and for these analyses the last step looks like figure 15.6.
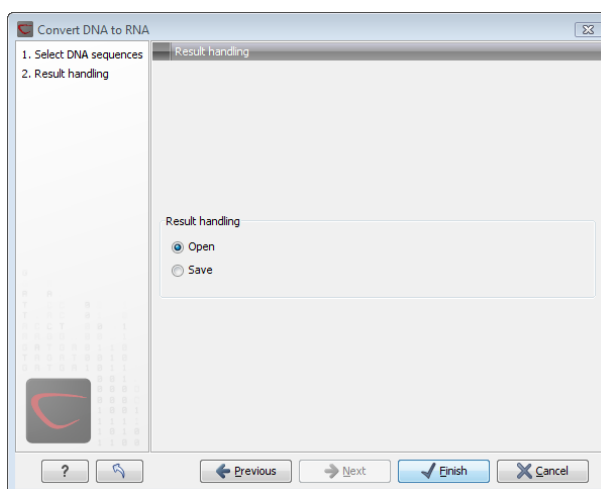
Figure 15.4: *The last step of the analyses exemplified by Translate DNA to RNA.*
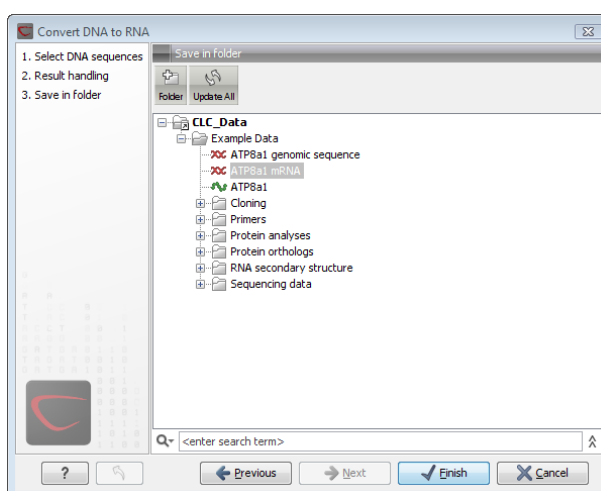
Figure 15.5: *Specify a folder for the results of the analysis.*

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**.

15.2.2 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 15.6). 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 15.7. In this example, the log displays information about how many open reading frames were found.

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.
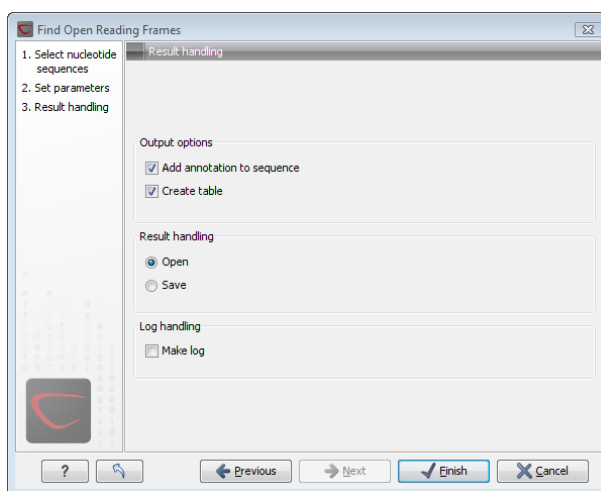
Figure 15.6: *Analyses which also generate tables.*

Figure 15.7: *An example of a batch log when finding open reading frames.*

15.3 Working with tables

Tables are used in a lot of places in the *CLC Main Workbench*. There are some general features for all tables, irrespective of their contents, that are described here.

Figure 15.8 shows an example of a typical table. This is the table result of **Find Open Reading Frames** (). We use this table as an example to illustrate concepts relevant to all kinds of tables.

Table viewing options

Options relevant to the view of the table can be configured in the **Side Panel** on the right.

For example, the columns that can be dispalyed in the table are listed in the section called **Show column**. The checkboxes allow you to see or hide any of the available columns for that table.

The Column width can be set to **Automatic** or **Manual**. By default, the first time you open a table, it will be set to **Automatic**. The default selected columns are hereby resized to fit the width of the viewing area. When changing to the **Manual** option, column widths will adjust to the actual header size, and each column size can subsequently by adjusted manually. When the table content exceeds the size of the viewing area, a horizontal scroll becomes available for navigation across the columns.

Sorting tables

You can **sort** table according to the values of a particular column by clicking a column header. (Pressing Ctrl - ⌘ on Mac - while you click will refine the existing sorting).

Clicking once will sort in ascending order. A second click will change the order to descending. A

Figure 15.8: *A table showing the results of an open reading frames analysis.*

third click will set the order back its original order.

15.3.1 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 15.9).[1]

Figure 15.9: *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.

---

[1]Note that for tables with more than 10000 rows, you have to actually click the **Filter** button for the table to take effect.

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:
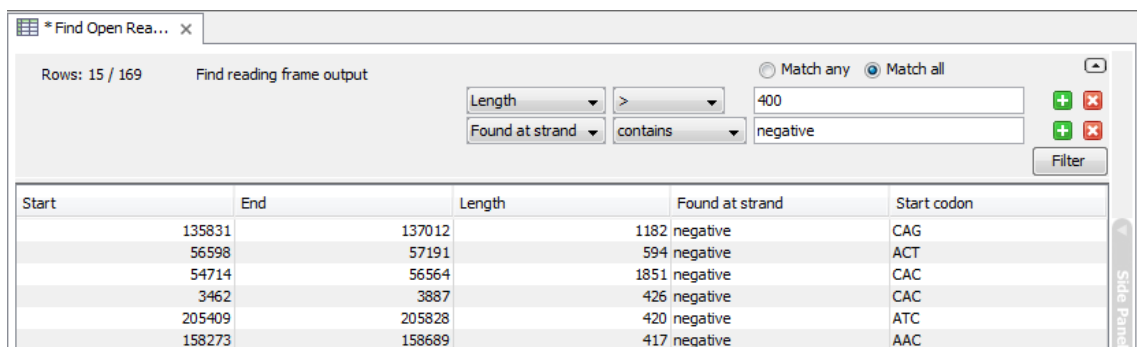
- **starts with** (the text starts with your search term)

- **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 15.10 shows an example of an advanced filter which displays the 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 15.9 and 15 in figure 15.10).

Figure 15.10: *The advanced filter showing open reading frames larger than 400 that are placed on the negative strand.*

Chapter 16

Workflows

Contents

The *CLC Main Workbench* provides a framework for creating, distributing, installing and running workflows. Workflows created in the Workbench can also be installed on a *CLC Genomics Server*.

A workflow consists of a series of connected tools where the output of one tool is used as input for another tool. In this way you create a workflow that for example makes a read mapping, uses the mapped reads as input for variant detection, and performs filtering of the variant track. Once the workflow is set up, it can be installed (either in your own Workbench or on a Server or it can be sent to a colleague). In that way it becomes possible to analyze lots of samples using the same standard pipeline, the same reference data and the same parameters.

This chapter will first explain how to create a new workflow, and next go into details about the installation and execution of a workflow. For information about installing a workflow on the *CLC Genomics Server*, please see the user manual at `http://www.clcbio.com/usermanuals`.

Note that the examples below are using tools from the *CLC Genomics Workbench* that are not available in the *CLC Main Workbench*. But the principles and workflow framework can be used in the same way with tools from *CLC Main Workbench*.

16.1 Creating a workflow

A workflow can be created by pressing the "Workflows" button (![icon]) in the toolbar and then selecting "New Workflow..." (![icon]).

Alternatively, a workflow can be created via the menu bar:

File | New | Workflow (![icon])

This will open a new view with a blank screen where a new workflow can be created.

16.1.1 Adding workflow elements

First, click the **Add Element** (![icon]) button at the bottom (or use the shortcut Shift + Alt + E). This will bring up a dialog that lists the elements and tools, which can be added to a workflow (see figure 16.1).

Alternatively elements can be dragged directly from the **Toolbox** into the workflow. Not all elements are workflow enabled. This means that only workflow enabled elements can be dropped in the workflow.
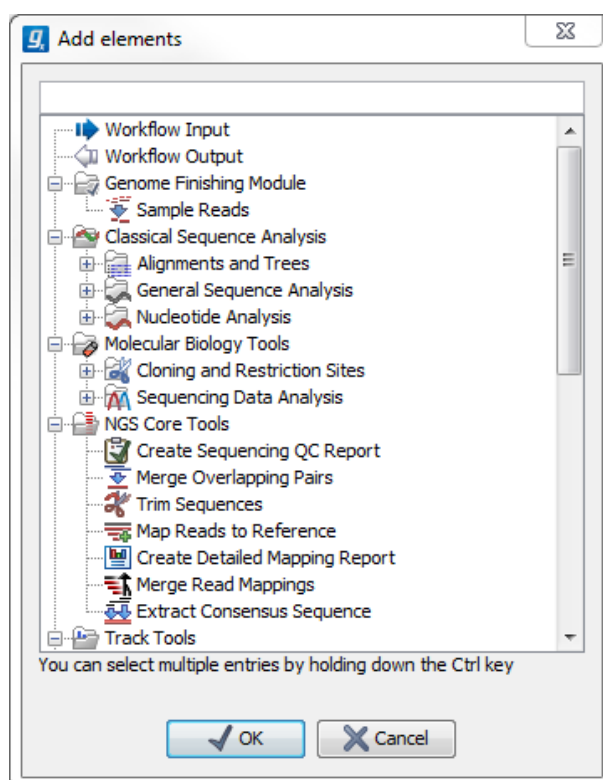
Figure 16.1: *Adding elements in the workflow.*

Elements that can be selected in the dialog are mostly tools from the Toolbox. However, there are two special elements on the list; the elements that are used for input and output. These two

elements are explained in section 16.1.5.

You can select more than one element in the dialog by pressing Ctrl (⌘ on Mac) while selecting. Click OK when you have selected the relevant tools (you can always add more later on).

You will now see the selected elements in the editor (see figure 16.2).
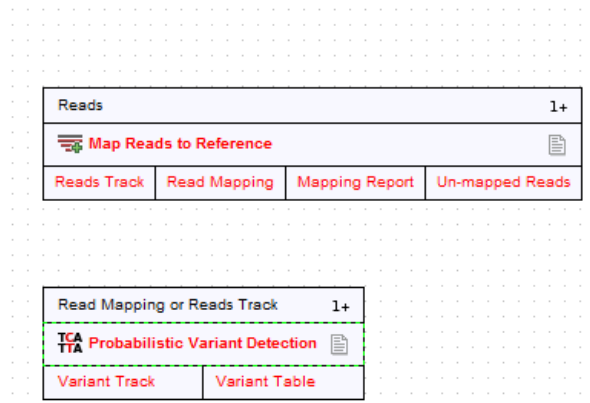
Figure 16.2: *Read mapping and variant calling added to the workflow.*

Once added, you can move and re-arrange the elements by dragging with the mouse (grab the box with the name of the element).

16.1.2 Configuring workflow elements

Each of the tools can be configured by right-clicking the name of the tool as shown in figure 16.3.
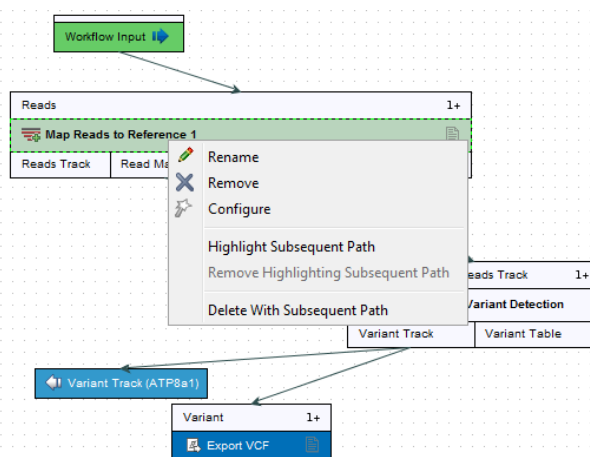
Figure 16.3: *Configuring a tool.*

The first option you are presented with is the option to **Rename** the element. This is for example useful when you wish to discriminate several copies of the same tool in a workflow. The name of the element is also visible as part of the process description when the workflow is executed. Right click on the tool in the workflow and select "Rename" or click on the tool in the workflow and use the F2 key as a shortcut.

With the **Remove** option, elements can be removed from the workflow. The shortcut Alt + Shift +

R removes all elements from the workflow.

You can also **Configure** the tool from the right click menu or alternatively it can be done by double-clicking the element. This will open a dialog with options for setting parameters, selecting reference data etc. An example is shown in figure 16.4.
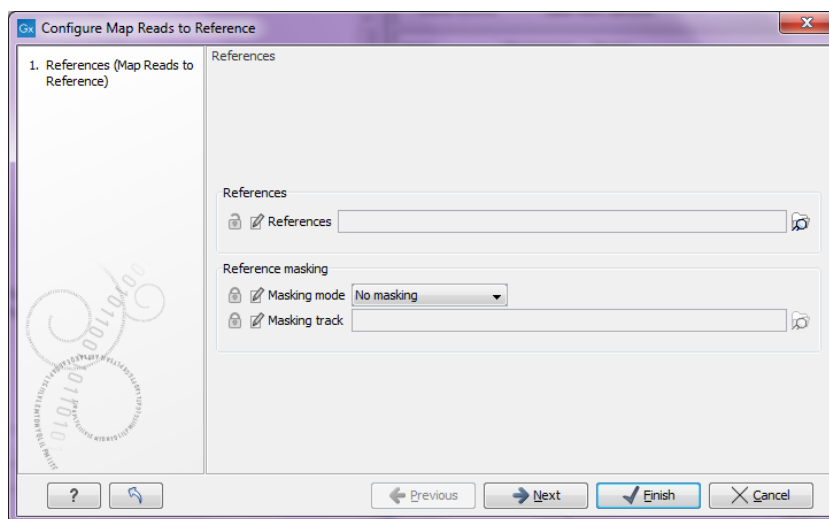
Figure 16.4: *Configuring read mapper parameters.*

Click through the dialogs using **Next** and press **Finish** when you are done. This will save the parameter settings that will then be applied when the workflow is executed.

You can also change the name of the parameter into something that fits the vocabulary of the users that are intended to execute the workflow. This is done by clicking the edit icon ( ) and entering a new name.

Note that reference data are a bit special. In the example with the read mapper in figure 16.3, you have to define a reference genome. This is done by pointing to data in the **Navigation Area**. If you distribute the workflow and install it in a different setting where this data is not accessible, the installation procedure will involve defining the new reference data to use (e.g. the reference genome sequence for read mapping). This is explained in more detail in section 16.2.

In some workflows, many elements use the same reference data, and there is a quick way of configuring all these: right-click the empty space and choose **Configure All References**. This will show a dialog listing all reference data needed by the workflow.

The lock icons in the dialog are used for specifying whether the parameter should be locked and unlocked as described in the next section.

Once an element has been configured, the workflow element gets a darker color to make it easy to see which elements have been configured.

With **Highlight Subsequent Path** the path from the tool that was clicked on and further downstream will be highlighted whereas all other elements will be grayed out (figure 16.5). The **Remove Highlighting Subsequent Path** reverts the highlighting to the normal workflow layout.

16.1.3 Locking and unlocking parameters

Figure 16.6 shows the different stages in a workflow.
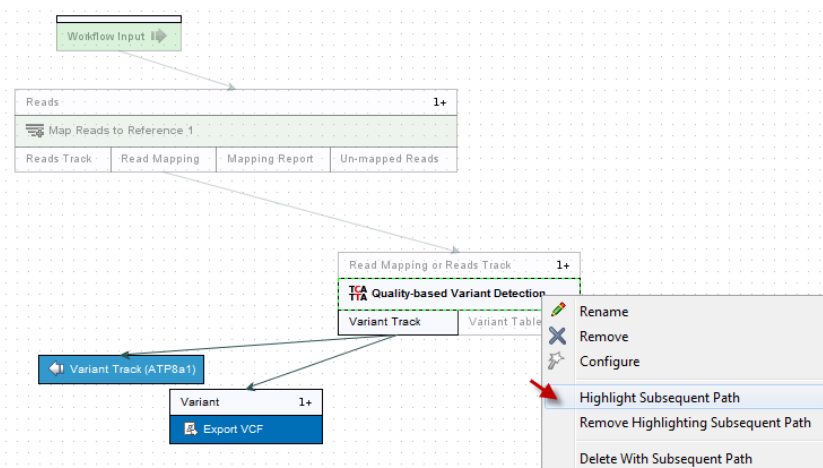
Figure 16.5: *Highlight path from the selected tool and downstream.*
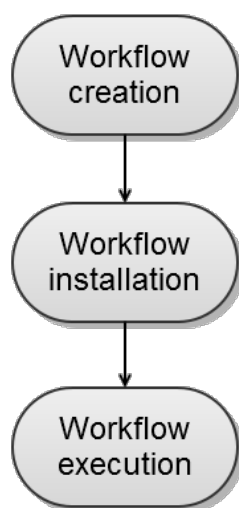
Figure 16.6: *The life cycle of a workflow.*

At the top, the workflow creation is illustrated. Workflow creation is explained above. Next, the workflow can be installed in a Workbench or Server (explained in section 16.2). Subsequently, the workflow can be executed as any other tool in the **Toolbox**.

At the creation step, the workflow creator can specify which parameters should be locked or unlocked. If a parameter is locked, it means that it cannot be changed neither in the installation nor the execution step. The lock icons shown in figure 16.4 specifies whether the parameter should be open or locked.

If the parameter is left open, it is possible to adjust it as part of the installation (see section 16.2). Furthermore, it can also be locked at this stage.

Parameters that are left open both from the workflow creation and installation, will be available for adjustment when the workflow is executed.

Please note that data parameters per default are marked as unlocked. When installing the workflow somewhere else, the connection to the data needs to be re-established, and this is only possible when the parameter is unlocked. Data parameters should only be locked if they should

not be set, or if the workflow will only be installed in a setting where there is access to the same
data.

16.1.4   Connecting workflow elements

Figure 16.7 explains the different parts of a workflow element.

Figure 16.7: *A workflow element consists of three parts: input, name of the tool, and output.*

At the top of each element a description of the required type of input is found. In the right-hand
side, a symbol specifies whether the element accepts multiple incoming connections, e.g. `+1`
means that more than one output can be connected, and no symbol means that only one can
be connected. At the bottom of each element there are a number of small boxes that represent
the different kinds of output that is produced. In the example with the read mapper shown in
figure 16.2, the read mapper is able to produce a reads track, a report etc.

Each of the output boxes can be connected to further analysis in three ways:

- By dragging with the mouse from the output into the input box of the next element. This is
  shown in figure 16.8. A green border around the box will tell you when the mouse button
  can be released, and an arrow will connect the two elements (see figure 16.9).

- Right-clicking the output box will display a list of the possible elements that this output
  could be connected to. You can also right-click the input box of an element and connect
  this to a matching output of another element.

- Alternatively, if the element to connect to is not already added, you can right-click the output
  and choose **Add Element to be Connected**. This will bring up the dialog from figure 16.1,
  but only showing the tools that accepts this particular output. Selecting a tool will both add
  it to the workflow and connect with the output you selected. You can also add an upstream
  element of workflow in the same way by right-clicking the input box.

Figure 16.8: *Dragging the reads track output with the mouse.*

All the logic of combining output and input is based on matching the type of input. So the read
mapper creates a reads track and a report as output. The variant caller accepts reads tracks
as input but not mapping reports. This means that you will not be able to connect the mapping
report to the variant caller.
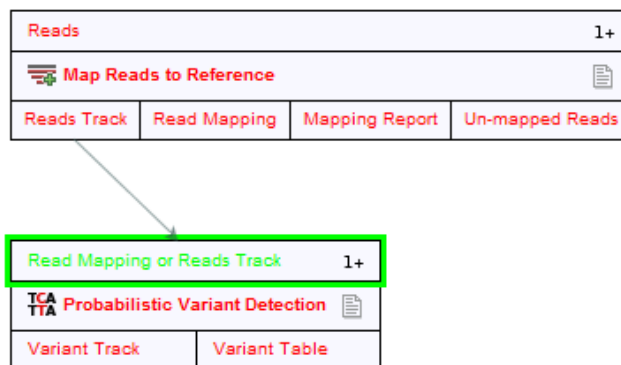
Figure 16.9: *The reads track is now used for variant calling.*

16.1.5   Input and output

Besides connecting the elements together, you have to decide what the output of the workflow should be. This is done by right-clicking the output box of any tool and selecting **Use as Workflow Output** as shown in figure 16.11.
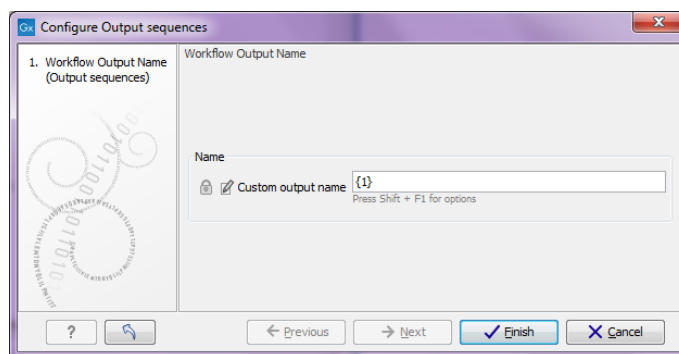
Figure 16.10: *Selecting a workflow output.*

You can mark several outputs this way throughout the workflow. Note that no intermediate results are saved unless they are marked as workflow output[1].

By double-clicking the output box, you can specify how the result should be named as shown in figure 16.11.

In this dialog you can enter a name for the output result, and you can make use of two dynamic placeholders for creating this name (press Shift + F1 to get assistance):

- {1} Represents the default name of the result. When running the tool outside of a workflow, this is the name given to the result.

- {2} Represents the name of the workflow input (not the input to this particular tool but the input to the entire workflow).

---

[1]When the workflow is executed, all the intermediate results are indeed saved temporarily but they are automatically deleted when the workflow is completed. If a part of the workflow fails, the intermediate results are not deleted.
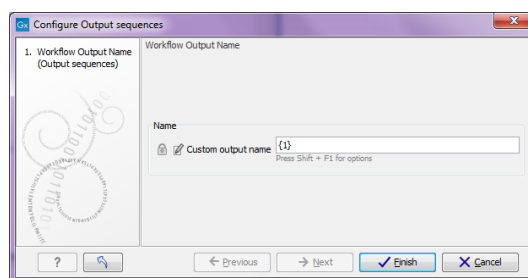
Figure 16.11: *Specifying nameing of a workflow output.*

An example of a meaningful name to a variant track could be `{2} variant track` as shown in figure 16.12. If your workflow input is named `Sample 1`, the result would be `Sample 1 variant track`.
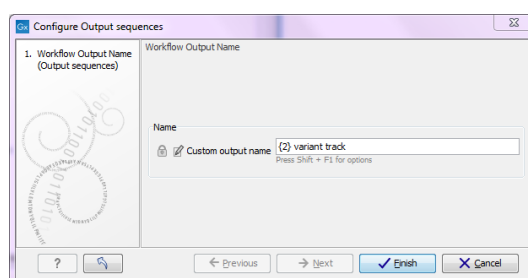
Figure 16.12: *Providing a custom name for the result.*

In addition to output, you also have to specify where the data should go into the workflow. When the workflow is executed, the user will provide some input data, and this has to be passed to the first element(s) in the workflow. This can be done by right-clicking the input box of the first tool and choose **Connect to Workflow Input**. By dragging from the workflow input box to other input boxes several tools can use the input data directly. Note that only one kind of input data will be provided as input, so you cannot specify e.g. both a mapping and a sequence list as input.

16.1.6   Layout

The workflow layout can be adjusted automatically. Right clicking in the workflow editor will bring up a pop-up menu with the option "Layout". Click on "Layout" to adjust the layout of the selected elements (Figure 16.13). Only elements that have been connected will be adjusted.

Note! The layout can also be adjusted with the quick command Shift + Alt + L.

Note! It is very easy to make an image of the workflow. Simply select the elements in the workflow (this can be done pressing Ctrl + A, by dragging the mouse around the workflow while holding down the left mouse button, or by right clicking in the editor and then selecting "Select All"), then press the Copy button in the toolbar ( ) or CTRL + C. Press Ctrl + V to paste the image into the wanted destination e.g. an email or a text or presentation program.
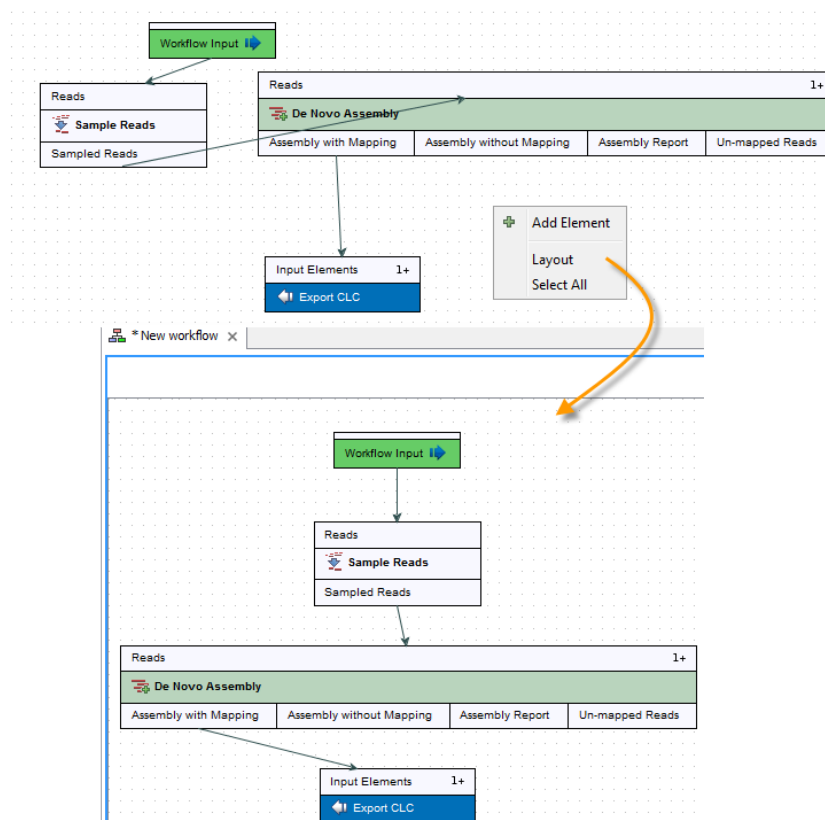
Figure 16.13: *A workflow layout can be adjusted automatically with the "Layout" function.*

16.1.7 Input modifying tools

An input modifying tool is a tool that manipulates its input objects (e.g. adds annotations) without producing a new object. This behavior differs from the rest of the tools and requires special handling in the workflow.

In the workflow an input modifying tool is marked with the symbol (Ⓜ) (figure 16.14).
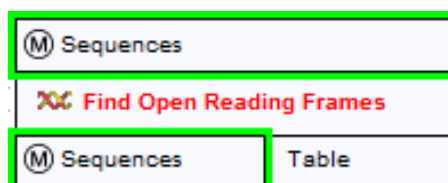
Figure 16.14: *Input modifying tools are marked with the letter M.*

Restrictions apply to workflows that contain input modifying tools. For example, branches are not allowed where one of the elements is a modifying tool (see figure 16.15), as it cannot be guaranteed which workflow branch will be executed first, which in turn means that different runs can result in production of different objects. Hence, if a workflow is constructed with a branch where one of the succeeding elements is a modifying tool, a message in red letters will appear saying "Branching before a modifying tool can lead to non-deterministic behavior". In such a situation the "Run" and "Create Installer" buttons will be disabled (figure 16.15).
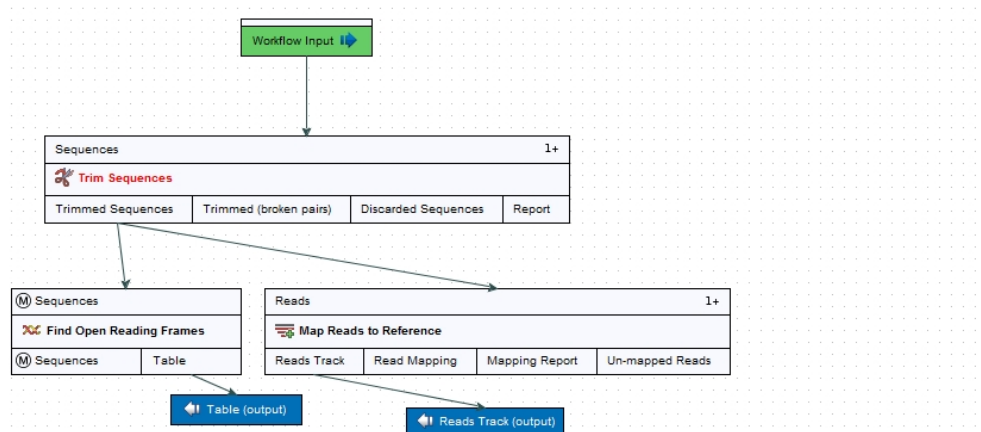
Figure 16.15: *A branch containing an input modifying tool is not allowed in a workflow.*

The problem can be solved by resolving the branch by putting the elements in the right order (with respect to order of execution). This is shown in figure 16.16 that also shows that the "Run" and "Create Installer" buttons are now enabled. In addition, a message in green letters has appeared saying "Validation successful".
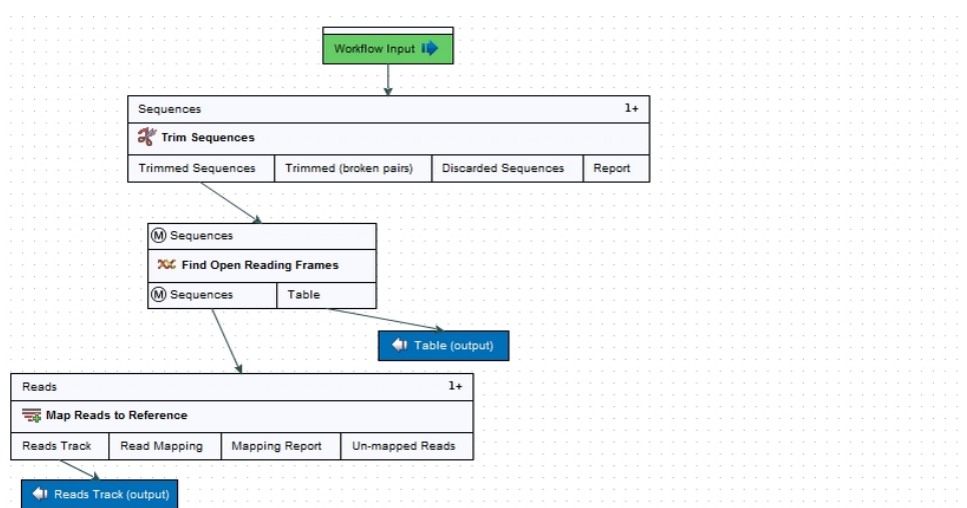
Figure 16.16: *A branch containing an input modifying tool has been resolved and the workflow can now be run or installed.*

As input modifying tools only modify existing objects without producing a new object, it is not possible to add a workflow output element directly after an input modifying tool (figure 16.17). A workflow output element can only be added when other tools than input modifying tools are included in the workflow.

If the situation occur where more input modifying tools are used succeedingly, a copy of the object will be created in addition to using the modified object as input at the next step of the chain (see figure 16.18). In order to see this output you must right click on the output option (marked with a red arrow in figure 16.18) and select "Use as Workflow Output".
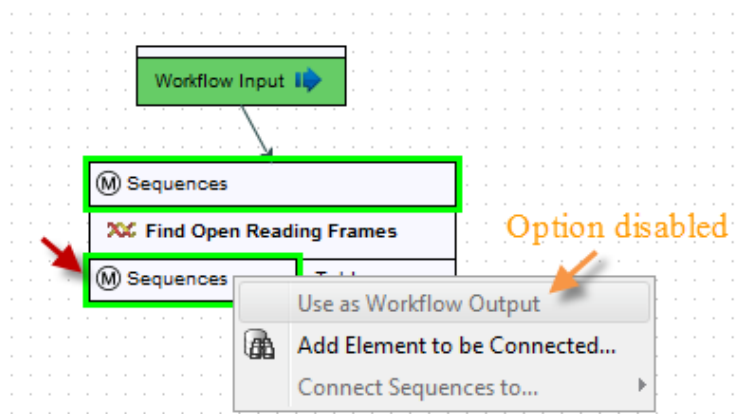
Figure 16.17: *A workflow output element cannot be added if the workflow only contains an input modifying tool.*
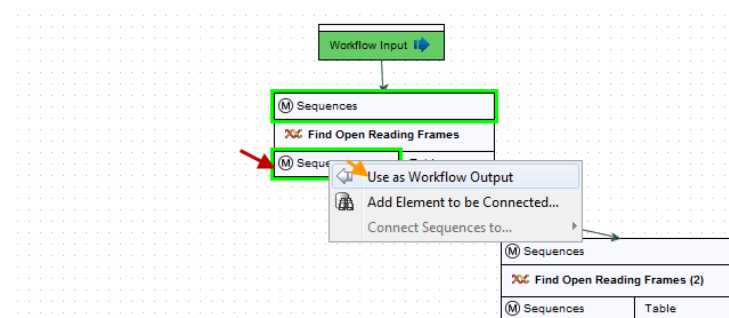
Figure 16.18: *A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Select "Use as Workflow Output" to make a copy of the output.*

When running a workflow where a workflow output has been added after the first input modifying tool in the chain (see figure 16.19) the output arrow is marked with "copy" to indicate that this is a copy of the result that is used as input at the next level in the chain. When running this workflow you will be able to see the copy of the output from the first input modifying tool in the **Navigation Area** (at the destination that you selected when running the workflow).
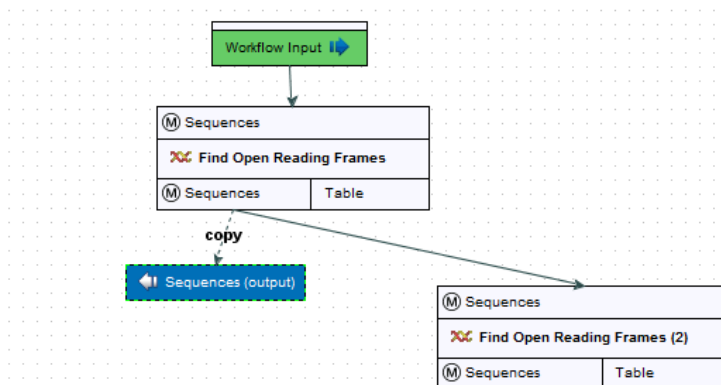
Figure 16.19: *A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Note that this output is marked with "copy" to indicate that this is a copy of the result that is used as input at the next level in the chain.*

16.1.8   Workflow validation

At the bottom of the view, there is a text with a status of the workflow (see figure 16.20). It will inform about the actions you need to take to finalize the workflow.
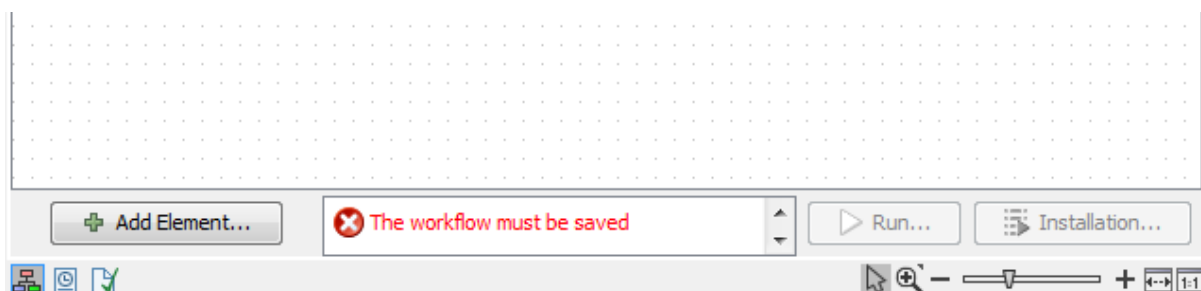
Figure 16.20: *A workflow is constantly validated at the bottom of the view.*

The validation may contain several lines of text. Scroll the list to see more lines. If one of the errors pertain to a specific element in the workflow, clicking the error will highlight this element.

The following needs to be in place before a workflow can be executed:

- All input boxes need to be connected either to the workflow input or to the output of other tools.

- At least one output box from each tool needs to be connected to either a workflow output or to the input box of another tool.

- The workflow has to be **Saved** (⬅).

- Additional checks that the workflow is consistent.

Once these conditions are fulfilled, the status will be "Validation successful", the **Run** button is enabled. Clicking this button will enable you to try running a data set through the workflow to test that it produces the expected results. If reference data has not been configured (see section 16.1.2), there will be a dialog asking for this as part of the test run.

16.1.9   Workflow creation helper tools

In the workflow editor **Side Panel**, you will find the following workflow display settings that can be useful to know (figure 16.21):

Grid

- Enable grid You can display a grid and control the spacing and color of the grid. Per default, the grid is shown, and the workflow elements snap to the grid when they are moved around.

View mode

- Collapsed The elements of the workflow can be collapsed to allow a cleaner view and especially for large workflows this can be useful.
- Highlight used elements Ticking **Highlight used elements** (or using the shortcut Alt + Shift + U) will show all elements that are used in the workflow whereas unused elements are grayed out.
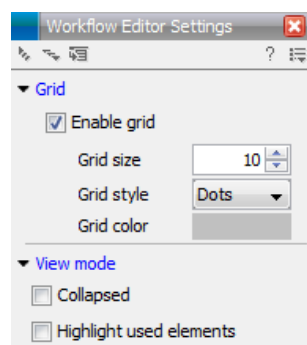
Figure 16.21: *The Side Panel of the workflow editor.*

16.1.10 Supported data flows

The current version of the workflow framework supports single-sample workflows. This means processing one sample through various analysis steps. When it comes to comparative analysis, this has to be done outside the workflow.

A typical example that would explain how this works is a trio analysis study where you want to compare variants found in a child with those from the mother and father. For this, you would create a workflow including mapping, variant detection, variant annotation and maybe some quality control. All three samples would be processed through this workflow in batch mode (see section 16.3). At the end, you can manually create a track list with all the relevant tracks (reads and variants) and run the trio analysis tool manually.

Since all the comparative tools are relatively quick, the bulk of the computation work can usually be incorporated into the workflow which can take care of the more tedious parts of the manual work involved.

CLC bio is planning further improvements to the workflow framework that allows you to model this kind of study as a workflow.

16.2 Distributing and installing workflows

Once the workflow has been configured, you can use the **Run** button (see section 16.1.8) to process data through the workflow, but the real power of the workflow is its ability to be distributed and installed in the **Toolbox** alongside the other tools that come with the *CLC Main Workbench*, as well as the ability to install the same workflow on a *CLC Genomics Server*. The mechanism for distributing the workflow is a workflow installer file which can be created from the workflow editor and distributed and installed in any Workbench or Server.

16.2.1 Creating a workflow installation file

At the bottom of the workflow editor, click the **Create Installer** button (or use the shortcut Shift + Alt + I) to bring up a dialog where you provide information about the workflow to be distributed (see an example with information from a CLC bio workflow in figure 16.22).

Author information Providing name, email and organization of the author of the workflow. This will be visible for users installing the workflow and will enable them to look up the source of the workflow any time. The organization name is important because it is part of the

Figure 16.22: *Workflow information for the installer.*

workflow id (see more in section 16.2.3)

Workflow name The workflow name is based on the name used when saving the workflow in the **Navigation Area**. The workflow name is essential because it is used as part of the workflow id (see more in section 16.2.3). The workflow name can be changed during the installation of the workflow. This is useful whenever you have a workflow that you would like to use e.g. with small variations. The original workflow name will remain the same in the **Navigation Area** - only the installed workflow will receive the customized name.

ID The final id of the workflow.

Workflow icon An icon can be provided. This will show up in the installation overview and in the **Toolbox** once the workflow is installed. The icon should be a 16 x 16 pixels gif or png file. If the icon is larger, it will automatically be resized to fit 16 x 16 pixels.

Workflow version A major and minor version can be provided.

Include original workflow file This will include the design file to be included with the installer. Once the workflow is installed in a workbench, you can extract the original workflow file and modify it.

Workflow description Provide a textual description of the workflow. This will be displayed for users when they have installed the workflow. Simple HTML tags are allowed (should be HTML 3.1 compatible, see `http://www.w3.org/TR/REC-html32`).

Click **Next** and you will be asked to specify where to install the workflow (figure 16.23). You can install your workflow directly on your local computer. If you are logged on a server and are the administrator, the option "Install the workflow on the current server" will be enabled. Finally, you can select to save the workflow as a .cpw file that can be installed on another computer. Click **Finish**. This will install the workflow directly on the selected destination. If you have selected to save the workflow for installation on another computer, you will be asked where to save the file after clicking **Finish**.
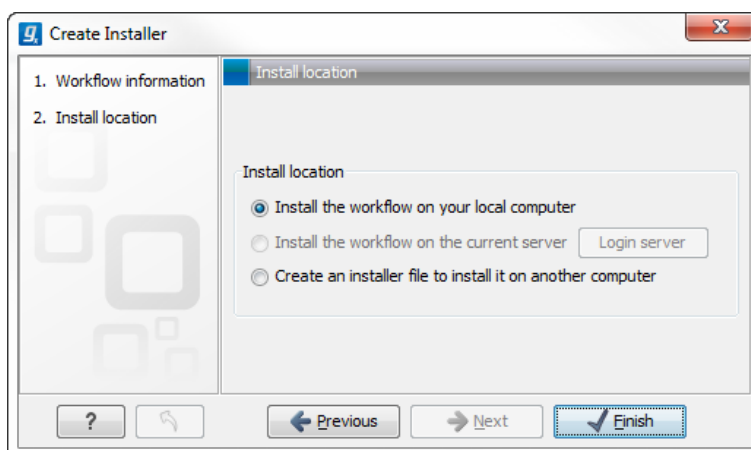
Figure 16.23: *Select whether the workflow should be installed on your local computer or on the current server. A third option is to create an installer file (.cpw) that can be installed on another computer.*

In cases where an existing workflow, that has already been installed, is modified, the workflow must be reinstalled. This can be done by first saving the workflow after it has been modified and then pressing the **Create Installer** button. Click through the wizard and select whether you wish to install the modified workflow on your local computer or on a server. Press **Finish**. This will open a pop-up dialog "Workflow is already installed" (figure 16.24) with the option that you can force the installation. This will uninstall the existing workflow and install the modified version of the workflow. **Note!** When forcing installation of the modified workflow, the configuration of the original workflow will be lost.
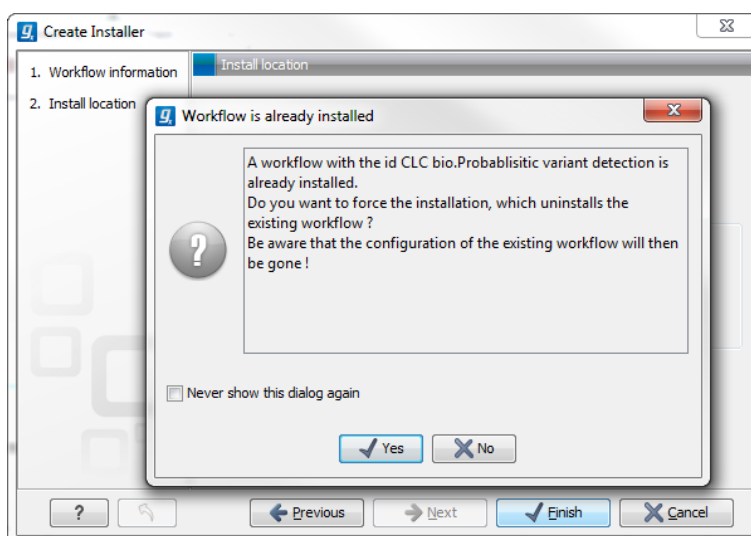
Figure 16.24: *Select whether you wish to force the installation of the workflow or keep the original workflow.*

16.2.2 Installing a workflow

Workflows are installed in the workflow manager (for information about installing a workflow on the *CLC Genomics Server*, please see the user manual at http://www.clcbio.com/usermanuals):

Help | **Manage Workflows** (⚏)

or press the "Workflows" button (⚏) in the toolbar and then select "Manage Workflow..." (⚏).

This will display a dialog listing the installed workflows. To install an existing workflow, click **Install from File** and select a workflow .cpw file .

Once installed, it will appear in the workflow manager as shown in figure 16.25.
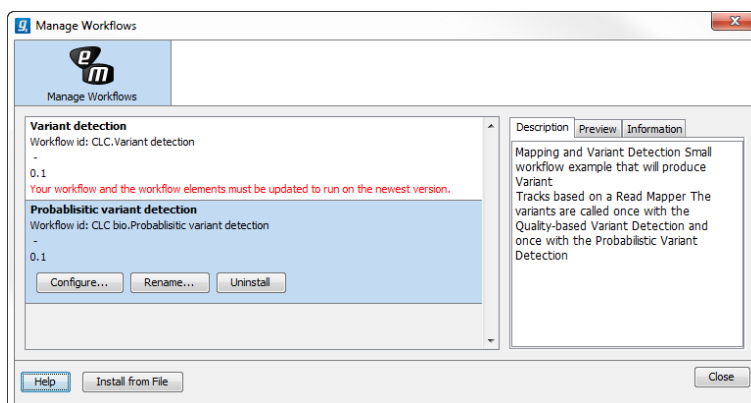
Figure 16.25: *Workflows available in the workflow manager. Note the alert on the "Variant detection" workflow, that means that this workflow needs to be updated.*

Click **Configure** and you will be presented with a dialog listing all the reference data that need to be selected. An example is shown in figure 16.26.
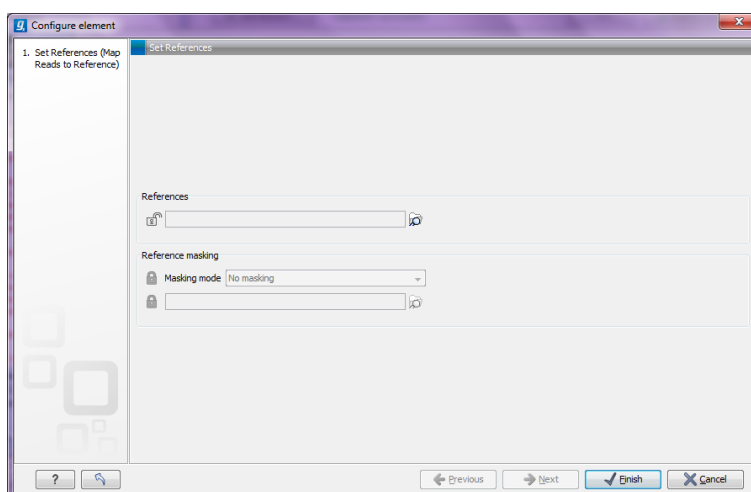
Figure 16.26: *Configuring parameters for the workflow.*

This dialog also allows you to further lock parameters of the workflow (see more about locking in section 16.1.3).

If the workflow is intended to be executed on a server as well, it is important to select reference data that is located on the server.

In addition to the configuration option, it is also possible to rename the workflow. This will change the name of the workflow in the **Toolbox**. The workflow id (see below) remains the same. To rename an element right click on the element name in the Navigation Area and select "Rename" or click on the F2 button.

In the right side of the window, you will find three tabs. **Description** contains the description that was entered when creating the workflow installer (see figure 16.22), the **Preview** shows a graphical representation of the workflow (figure 16.27), and finally you can get **Information** about the workflow (figure 16.28).
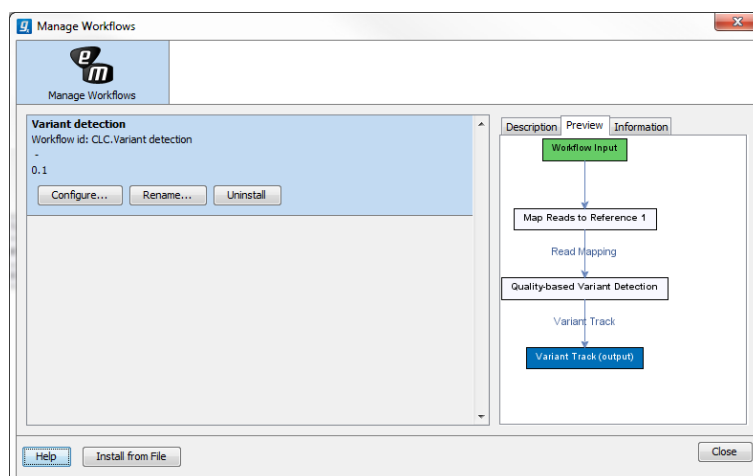
Figure 16.27: *Preview of the workflow.*

The "Information" field (figure 16.28) contains the following:

Build id The date followed by the time

Download href The name of the workflow .cpw file

Id The unique id of a workflow, by which the workflow is identified

Major version The major version of the workflow

Minor version The minor version of the workflow

Name Name of workflow

Rev version Revision version. The functionality is activated but currently not in use

Vendor id ID of vendor that has created the workflow

Version <Major version>.<Minor version>

Workbench api version Workbench version

Workflow api version Workflow version (a technical number that can be used for troubleshooting)

16.2.3   Workflow identification and versioning

A workflow has a version. The version is used to make it easy to distribute an improved version of the same workflow. To do this, create a new installer with an incremented version number. In order to install a new and updated version, the old one has to be uninstalled.

The way the *CLC Main Workbench* checks whether a workflow already exists in a previous version is by looking at the workflow id. The id is a combination of the organization name
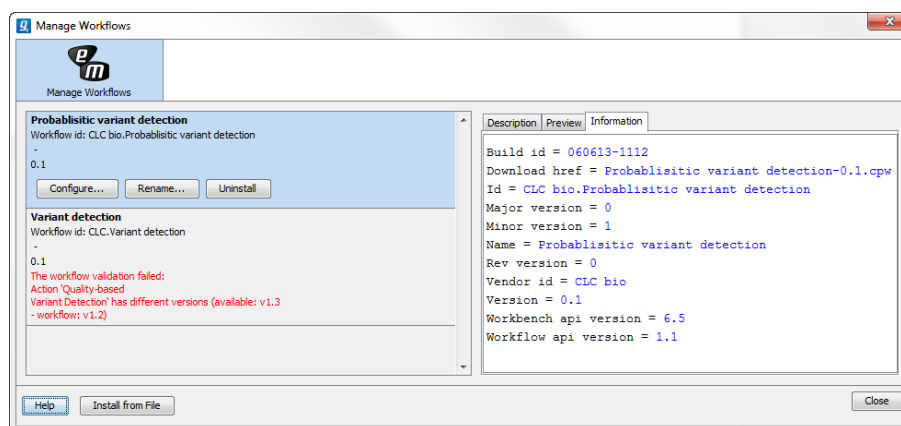
Figure 16.28: *With "Manage Workflows" it is possible to configure, rename and uninstall workflows.*

and the name of the workflow itself as it is shown in the dialog shown in figure 16.22. Once installed this information is also available in the workflow manager (in figure 16.25 this is `CLC bio.Simple variant detection and annotation-1.2`).

If you create two different workflows with the same name and using the same organization name when creating the installer, they cannot both be installed.

16.2.4   Automatic update of workflow elements

When new versions of the *CLC Main Workbench* are released, some of the tools that are part of a workflow may change. When this happens, the workflow may no longer be valid. This will happen both to the workflow configurations saved in the **Navigation Area** and the installed workflows.

When a workflow is opened from the **Navigation Area**, an editor will appear, if tools used in the workflow have been updated (see figure 16.29).
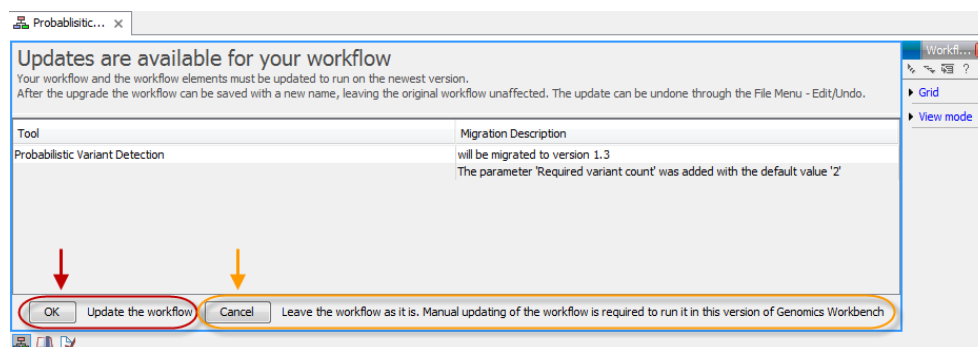
Figure 16.29: *When updates are available an editor appears with information about which tools should be updated. Press "OK" to update the workflow. The workflow must be updated to be able to run the workflow on the newest version of the Workbench.*

Updating a workflow means that the tools in your workflow is updated with the most recent version of these particular tools. To update your workflow, press the **OK** button at the bottom of the page.

There may be situations where it is important for you to keep the workflow in its original form. This could be the case if you have used a workflow to generate results for a publication. In such cases it may be necessary for you to be able to go back to the original workflow to e.g. repeat an analysis.

You have two options to keep the old workflow:

- If you do not wish to update the workflow at all, press the **Cancel** button. This will keep the workflow unchanged. However, the next time you open the workflow, you will again be asked whether you wish to update the workflow. Please note that only updated workflows can run on the newest versions of the Workbench.

- Another option is to update the workflow and save the updated workflow with a new name. This will ensure that the old workflow is kept rather than being overwritten.

Note! In cases where new parameters have been added, these will be used with their default settings.

If you have used the toolbar "Workflow" button (🖳) and "Manage Workflow..." (🖳) to access a specific workflow in order to e.g. change the workflow configuration or are going to use the "Install from File" function, a button labeled "Update..." will appear whenever tools have been changed and the workflow needs to be updated (figure 16.30). When you click the button labeled "Update...", your workflow will be updated and the existing workflow will be overwritten.
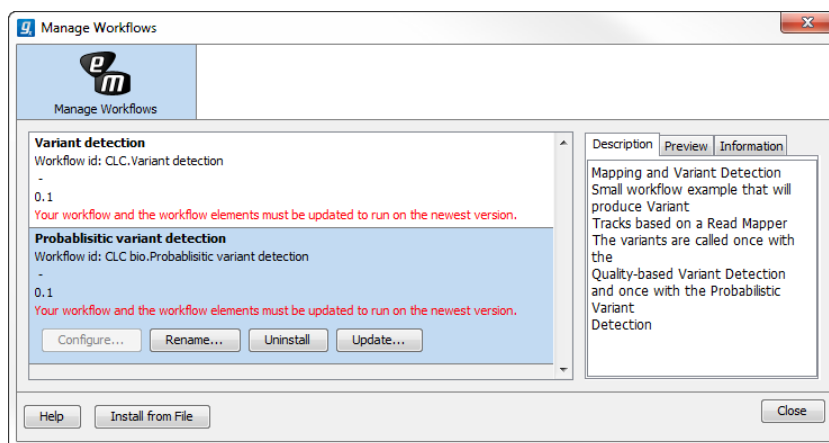
Figure 16.30: *Workflow migration.*

16.3   Executing a workflow

Once installed and configured, a workflow will appear in the **Toolbox** under **Workflows** (📷). If an icon was provided with the workflow installer this will also be shown (see figure 16.31).
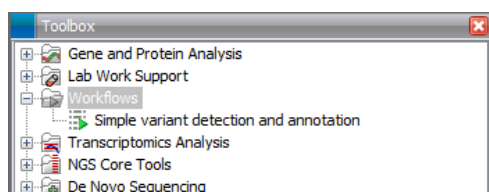
Figure 16.31: *A workflow is installed and ready to be used.*

The workflow is executed just as any other tool in the **Toolbox** by double-clicking or selecting it in the menu (or with the shortcut Ctrl + Enter). This will open a dialog where you provide input data and with options to run the workflow in batch mode (see section 15.1).

If you are connected to a *CLC Genomics Server*, you will be presented with the option to run the workflow locally on the Workbench or on the Server. When you are selecting where to run the workflow, you should also see a message should there be any missing configurations. There are more details about running Workflows on the Server in the Server manual (`http://www.clcsupport.com/clcgenomicsserver/current/admin/index.php?manual=Workflows.html`).

When the workflow is started, you can see the log file with detailed information from each step in the process.

If the workflow is not properly configured, you will see that in the dialog when the workflow is started [2].

---

[2]If the workflow uses a tool that is part of a plugin, a missing plugin can also be the reason why the workflow is not enabled. A workflow can also become outdated because the underlying tools have changed since the workflow was created (see section 16.2.3)

Part III

Bioinformatics

Chapter 17

Viewing and editing sequences

Contents

CLC Main 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.

17.1 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 9.2 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 29.2).

17.1.1 Sequence settings in Side Panel

Each view of a sequence has a **Side Panel** located at the right side of the view (see figure 17.1.
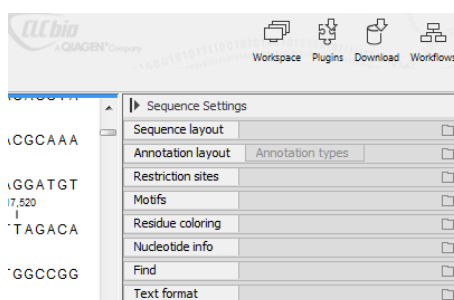
Figure 17.1: *Overview of the Side Panel which is always shown to the right of a view.*

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 (◀▌) 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 11.6 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).

- **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).

- **Matching residues as dots** Residues in aligned sequences identical to residues in the first (reference) sequence will be presented as dots. An option that is only available for "Alignments" and "Read mappings".

Annotation Layout and Annotation Types

See section 17.3.2.

Restriction sites

See section 17.1.2.

Motifs

See section 22.8.1.

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 following categories:

  - **Green** neutral, polar
  - **Black** neutral, nonpolar
  - **Red** acidic, polar
  - **Blue** basic ,polar
  - As with other options, you can choose to set or change the coloring for either the residue letter or its background:

    * **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.

- **Color space encoding.** Lets you define a few settings for how the colors should appear.

  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 **??**.

Hide unaligned 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.

- **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).
    * **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 17.1.3.
    * **+1 to -1.** Select one of the six reading frames.
    * **All forward/All reverse.** Shows either all forward or all reverse reading frames.
    * **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 23.5.
  - **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 26.1.

- **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 13.4).
    * **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 13.4).

    * **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.

- **Secondary structure.** Allows you to choose how to display a symbolic representation of the secondary structure along the sequence (See section 31.2.3 for a detailed description of the settings).

Protein info

These preferences only apply to proteins. The first nine items are different hydrophobicity scales and are described in section 24.5.2.

- **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 be used for searching the sequence and is invoked by pressing Ctrl + Shift + F (⌘ + Shift + F on Mac). Initially, specify the 'search term' to be found, select the type of search (see various options in the following) and finally click on the Find button. The first occurrence of the search term will then be highlighted. 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 or number 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 17.3.3). If you enter positions including thousands separators like 123,345, the comma will just be ignored and it would be equivalent to entering 123345.

- **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.

17.1.2 Restriction sites in the Side Panel

Please see section 28.3.1.

17.1.3   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 17.1.2.)

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 Analysis ( )| 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.

17.1.4 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 28.1.4.

Note When editing annotated nucleotide sequences, the annotation content is not updated automatically (but its position is). Please refer to section 17.3.4 for details on annotation editing. Before exporting annotated nucleotide sequences in GenBank format, ensure that the annotations in the Annotations Table reflect the edits that have been made to the sequence.

17.1.5 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 17.2 is an example of three regions with separate colors.
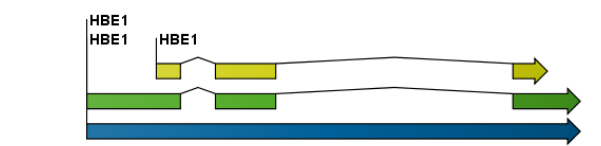
Figure 17.2: *Three regions on a human beta globin DNA sequence (HUMHBB).*

Figure 17.3 shows an artificial sequence with all the different kinds of regions.

17.2 Circular DNA

A sequence can be shown as a circular molecule:

> **Select a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Select "Circular View" ( )**
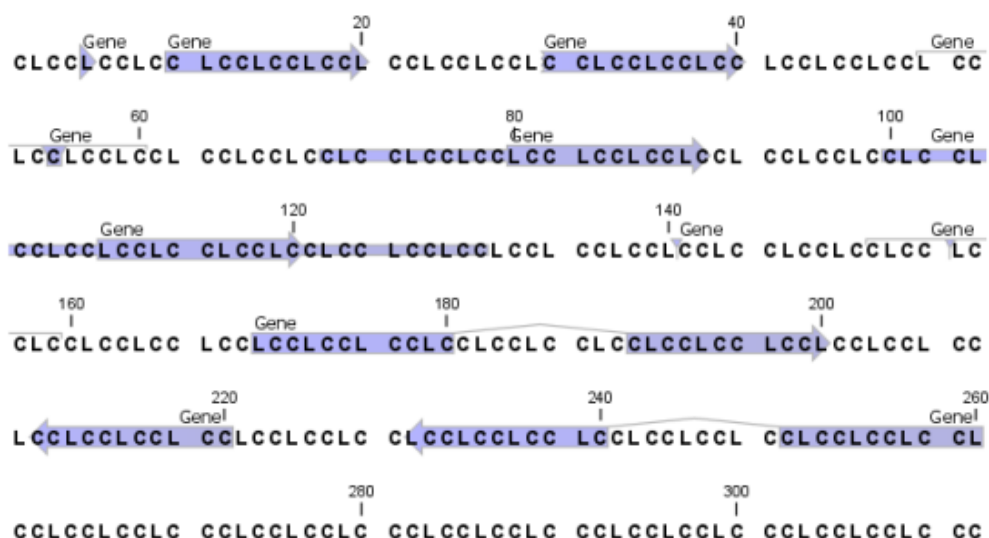
Figure 17.3: *Region #1: A single residue, Region #2: A range of residues including both endpoints, Region #3: A range of residues starting somewhere before 30 and continuing up to and including 40, Region #4: A single residue somewhere between 50 and 60 inclusive, Region #5: A range of residues beginning 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.*

   or   **If the sequence is already open | Click "Show Circular View" (🔴) at the lower left part of the view**

This will open a view of the molecule similar to the one in figure 17.4.
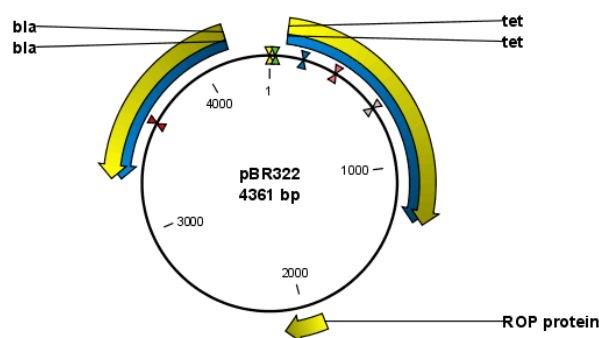
Figure 17.4: *A molecule shown in a circular view.*

This view of the sequence shares some of the properties of the linear view of sequences as described in section 17.1, 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.

17.2.1 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 (ᴬᴳᵀ) 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 17.5.
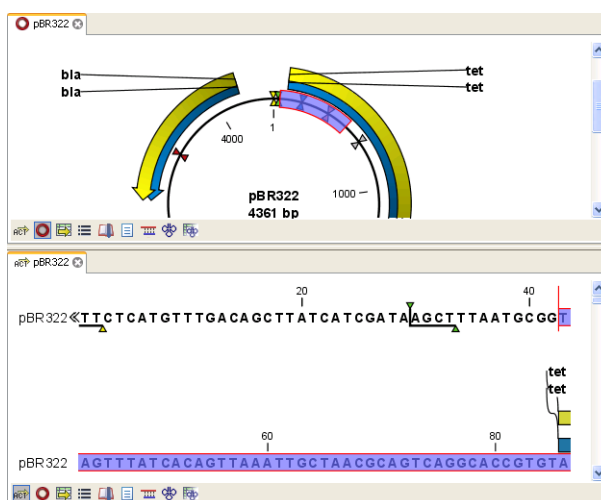
Figure 17.5: *Two views showing the same sequence. The bottom view is zoomed in.*

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.

17.2.2 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 ».

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.

17.3   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 Main Workbench*, sequences can have annotations (GenBank, EMBL and Swiss-Prot format).

- The result of a number of analyses in *CLC Main 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 17.3.3).

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.

17.3.1   Extract Annotations

The **Extract annotations** tool makes it very easy to extract parts of a sequence (or several sequences) based on its annotations. Using a few steps it is possible to:

- extract e.g. all tRNA genes from the *E. coli* genome.

- automatically add flanking regions to the annotated sequences.

- search for specific words in all available annotations.

The output is a sequence list that contains sequences carrying the annotation specified (including the flanking regions, if this option was selected).

To extract annotations from a sequence:

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

This opens the dialog shown in figure 17.6 that allows specification of which sequence to extract annotations from.

Click **Next**. At the top of the dialog shown in figure 17.7 you can specify which annotations to use:

- **Search term**. All annotations and attached information for each annotation will be searched for the entered term. It can be used to make general searches for search terms such as "Gene" or "Exon", or it can be used to make more specific searches. If you e.g. have a gene
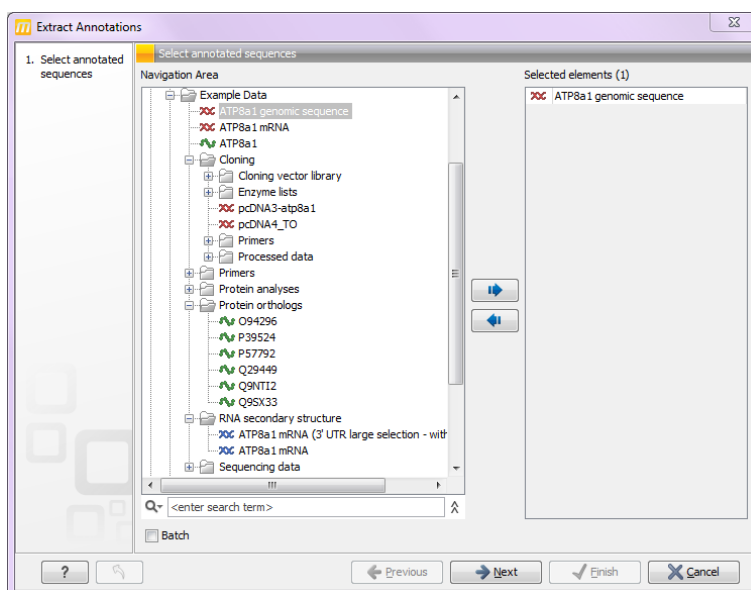
Figure 17.6: *Select one or more sequences to extract annotations from.*
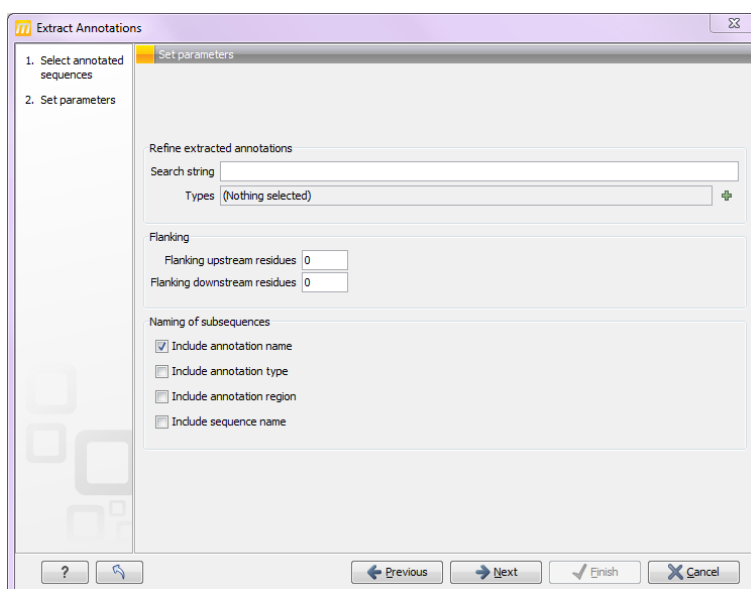
Figure 17.7: *Adjusting parameters for extract annotations.*

annotation called "MLH1" and another called "MLH3", you can extract both annotations by entering "MLH" in the search term field. If you wish to enter more specific search terms, separate them with commas, e.g. "MLH1, Human" will find annotations including both "MLH1" and "Human".

- **Annotation types** If only certain types of annotations should be extracted, this can be specified here.

The sequence of interest can be extracted with flanking sequences:

- **Flanking upstream residues.** The output will include this number of extra residues at the 5' end of the annotation.

- **Flanking downstream residues.** The output will include this number of extra residues at the 3' end of the annotation.

The sequences that are created can be named after the annotation name, type etc:

- **Include annotation name.** This will use the name of the annotation in the name of the extracted sequence.

- **Include annotation type.** This corresponds to the type chosen above and will put this information in the name of the resulting sequences. This is useful information if you have chosen to extract "All" types of annotations.

- **Include annotation region.** The region covered by the annotation on the original sequence (i.e. not including flanking regions) will be included in the name.

- **Include sequence/track name.** If you have selected more than one sequence as input, this option enables you to discern the origin of the resulting sequences in the list by putting the name of the original sequence into the name of the resulting sequences.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

17.3.2 Viewing annotations

Annotations can be viewed in a number of different ways:

- As arrows or boxes in all views displaying sequences (sequence lists, alignments etc)

- 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 17.8 shows an annotation displayed on a sequence.
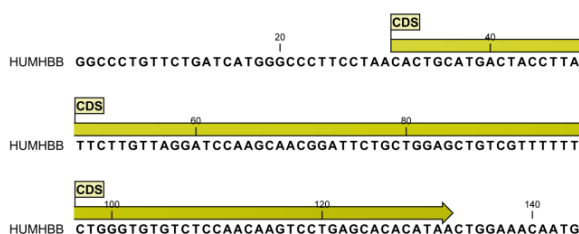
Figure 17.8: *An annotation showing a coding region on a genomic dna sequence.*

The various sequence views listed in section 17.3.2 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 17.9.
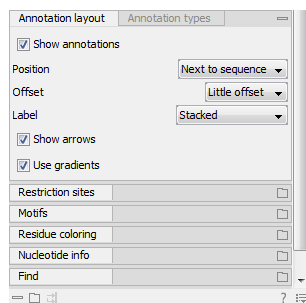
Figure 17.9: *The annotation layout in the Side Panel. The annotation types can be shown by clicking on the "Annotation types" tab.*

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).

- **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 17.10).
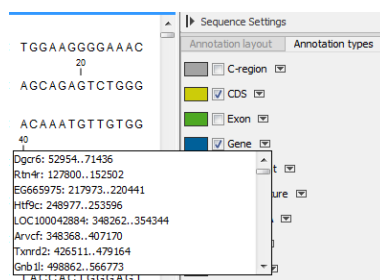
Figure 17.10: *Browsing the gene annotations on a sequence.*

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 a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | 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 17.11).

In the **Side Panel** you can show or hide individual annotation types in the table. E.g. if you only wish to see "gene" annotations, de-select the other annotation types so that only "gene" is selected.

Figure 17.11: *A table showing annotations on the sequence.*

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 17.3.3).

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 17.3.3).

17.3.3   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 a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Annotation table ( ) | right click anywhere in the annotation table | select Add Annotation ( )**

This will display a dialog like the one in figure 17.12.
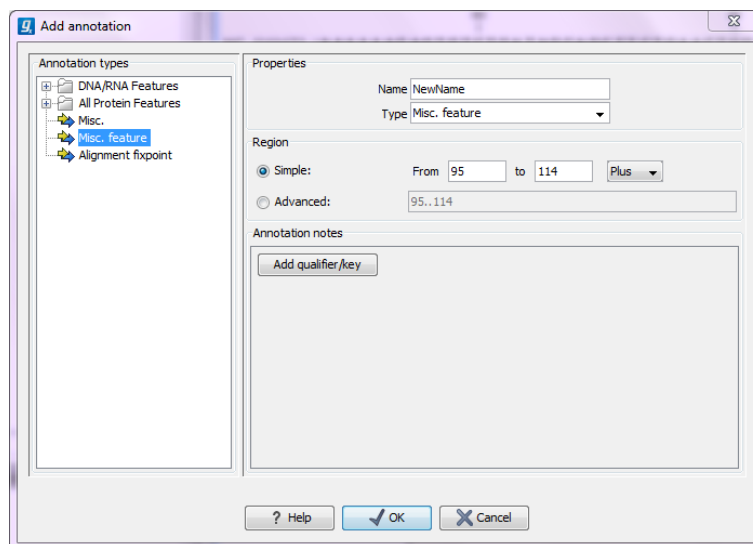
Figure 17.12: *The Add Annotation dialog.*

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 17.3.2).

- **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.

---

[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

- **<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 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 17.3.2) 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.

17.3.4   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 17.12, 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 17.13.
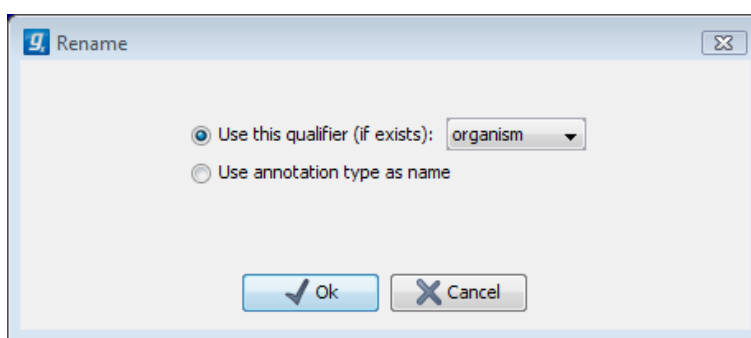
Figure 17.13: *The Advanced Rename dialog.*

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 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 17.14.

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.
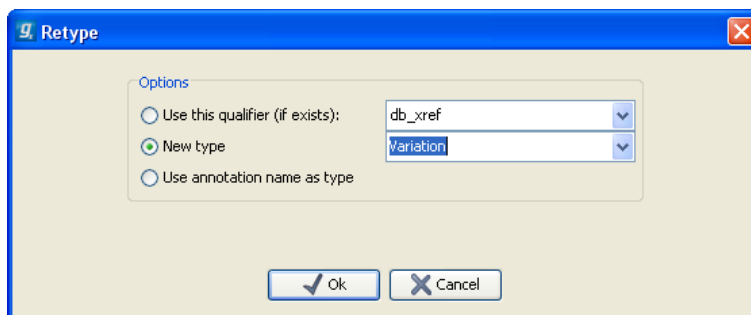
Figure 17.14: *The Advanced Retype dialog.*

- **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.

- **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).

17.3.5   Removing annotations

Annotations can be hidden using the **Annotation Types** preferences in the **Side Panel** to the right of the view (see section 17.3.2). In order to completely remove the annotation:

>   **right-click the annotation | 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**

17.4   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 and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Element Info ( )**

Another way to show the text view is to open the sequence in the **View Area** and click on the

"Show Element Info" icon ( ) found at the bottom of the window.

This will display a view similar to fig 17.15.

Figure 17.15: *The initial display of sequence info for the HUMHBB DNA sequence from the Example data.*

All the lines in the view are headings, and the corresponding text can be shown by clicking the text.

- **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 14) for information about the latest changes to the sequence after it was downloaded from the database.

- **Latin name.** Latin name of the organism.

- **Common name.** Scientific name of the organism.

- **Taxonomy name.** Taxonomic classification levels.

The information available depends on the origin of the sequence. Sequences downloaded from database like NCBI and UniProt (see section 18) 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**.

17.5 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 and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Select "Text View" (▤)**

Another way to show the text view is to open the sequence in the **View Area** and click on the "Show Text View" icon (▤) found at the bottom of the window.

This makes it 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 17.4.)

In the **Side Panel**, you find a search field for searching the text in the view.

17.6 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.

Sequence lists are generated automatically when you import files containing more than one sequence. Sequence lists may also be created as the output from particular Workbench tools, including database searches such as a (see (chapter 18.1)).

Sequence List can also be created from other sequences or sequence lists within the Workbench. To do this:

> **select two or more sequences or sequence lists | right-click the elements | New | Sequence List (≣)**

Alternatively, you can launch this took via the menu system:

> **File | New | Sequence List (≣)**

This opens the **Sequence List** Wizard:

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**.
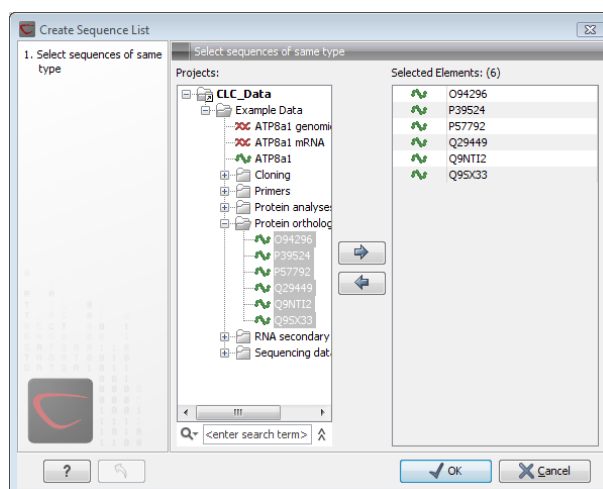
Figure 17.16: *A Sequence List dialog.*

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 17.17.
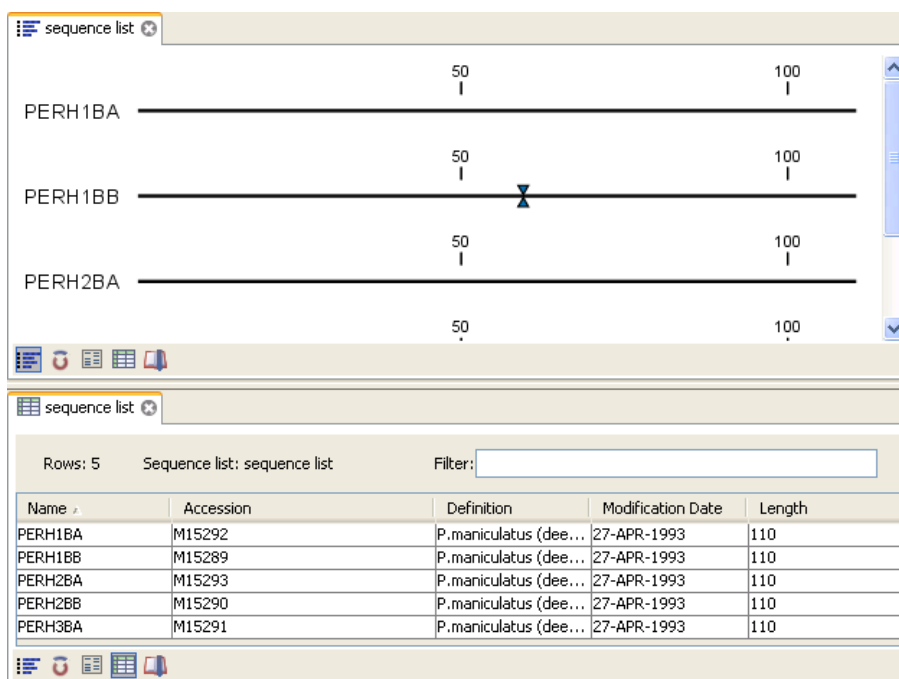
Figure 17.17: *A sequence list containing multiple sequences can be viewed in either a table or in a graphical sequence list. The graphical view is useful for viewing annotations and the sequence itself, while the table view provides other information like sequence lengths, and the number of sequences in the list (number of Rows reported).*

17.6.1 Graphical view of sequence lists

The graphical view of sequence lists is almost identical to the view of single sequences (see section 17.1). 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**.

17.6.2 Sequence list table

Each sequence in the table sequence list is displayed with:

- Name.

- Accession.

- Description.

- Modification date.

- Length.

- First 50 residues.

The number of sequences in the list is reported as the number of Rows at the top of the table view.

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.

Learn more about tables in Appendix 15.3.

17.6.3 Extract sequences from sequence list

Sequences can be extracted from a sequence list when the sequence list is opened in tabular view. One or more sequences can be dragged (with the mouse) directly from the table 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 be done with the **Extract Sequences** tool:

> **Toolbox | General Sequence Analysis (** **)| Extract Sequences (** **)**   .

A description of how to use the **Extract Sequences** tool can be found in section 22.1.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Chapter 18

Data download

Contents

CLC Main Workbench offers different ways of searching and downloading online data. You must be online when initiating and performing the following searches:

18.1 GenBank search

This section describes searches for sequences in GenBank - the **NCBI Entrez** database. The NCBI search view is opened in this way (figure 18.1):

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

> or **Ctrl + B (⌘ + B on Mac)**

This opens the following view:
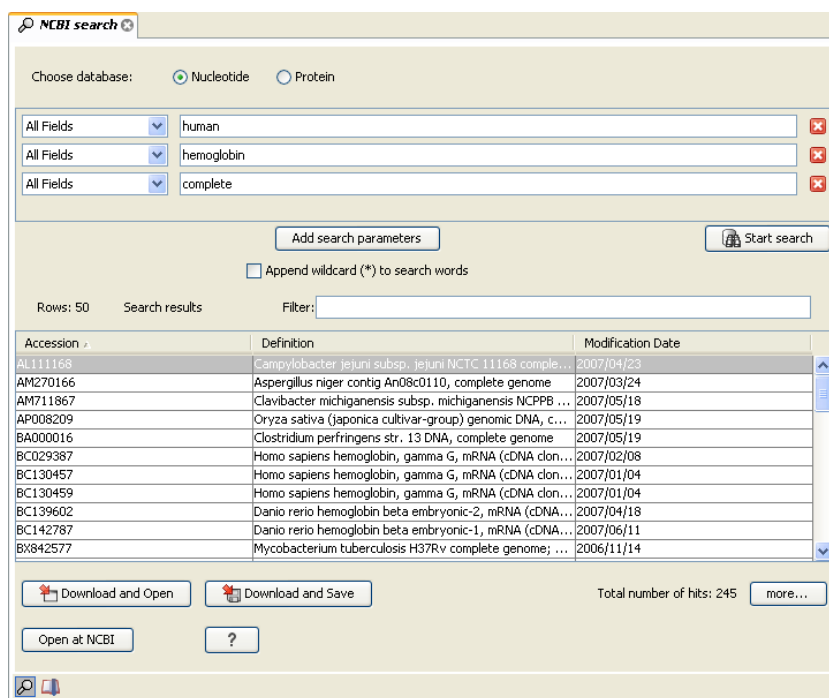
Figure 18.1: *The GenBank search view.*

18.1.1   GenBank search options

Conducting a search in the **NCBI Database** from *CLC Main Workbench* corresponds to conducting the search on NCBI's website. When conducting the search from *CLC Main 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 Main 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/books/NBK3837/

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.

18.1.2 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 11). 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 11.6.

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 18.2). 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 18.2: *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.

18.1.3   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.

18.2   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 18.3) is opened in this way:

Download | Search for Sequences in UniProt ( )
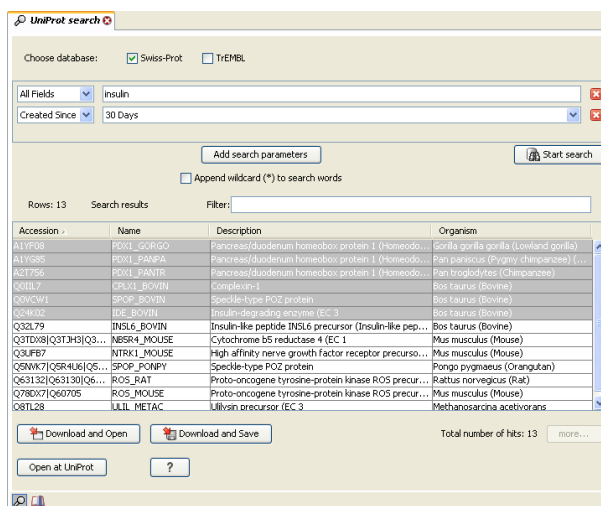
Figure 18.3: *The UniProt search view.*

18.2.1   UniProt search options

Conducting a search in **UniProt** from *CLC Main Workbench* corresponds to conducting the search on UniProt's website. When conducting the search from *CLC Main 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 Main 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.

18.2.2   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 11). 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 11.6.

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 18.2). 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.

18.2.3 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.

18.3 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 21.2.

The NCBI search view is opened in this way:

> **Download | Search for structures at NCBI ( )**

or   **Ctrl + B (⌘ + B on Mac)**

This opens the view shown in figure 18.4:
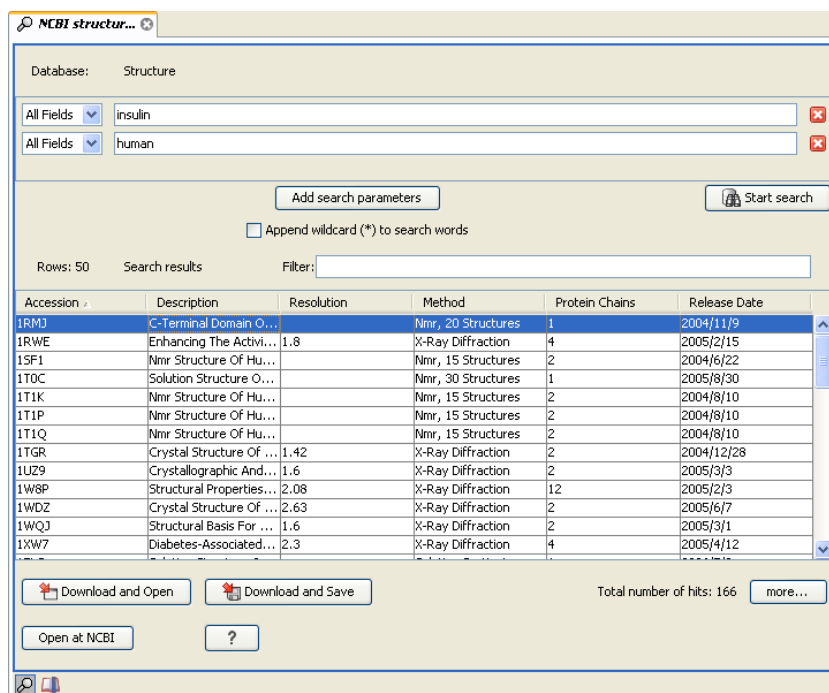
Figure 18.4: *The structure search view.*

18.3.1   Structure search options

Conducting a search in the **NCBI Database** from *CLC Main Workbench* corresponds to conducting search for structures on the NCBI's Entrez website. When conducting the search from *CLC Main Workbench*, the results are available and ready to work with straight away.

As default, *CLC Main 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 Gen-Bank 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.

18.3.2   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 11). 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 11.6.

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 18.5). 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.
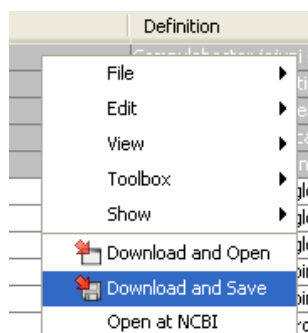
Figure 18.5: *By right-clicking a search result, it is possible to choose how to handle the relevant structure.*

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 PDB 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.

18.3.3   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.

18.4 Sequence web info

CLC Main 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 17.5). 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 18.6):

Open a sequence or a sequence list | Right-click the name of the sequence | Web Info ( ) | select the desired search function
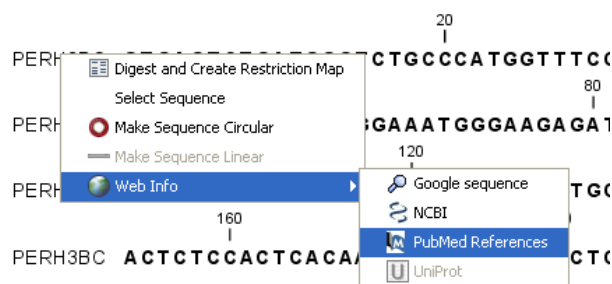
Figure 18.6: *Open webpages with information about this sequence.*

This will open your computer's default browser searching for the sequence that you selected.

18.4.1 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.

18.4.2 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.

18.4.3   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.

18.4.4   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.

18.4.5   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. For tracks, these links are also available in the track table.

Chapter 19

BLAST search

Contents

CLC Main 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 19.5.

Figure 19.9 shows an example of a BLAST result in the *CLC Main Workbench*.
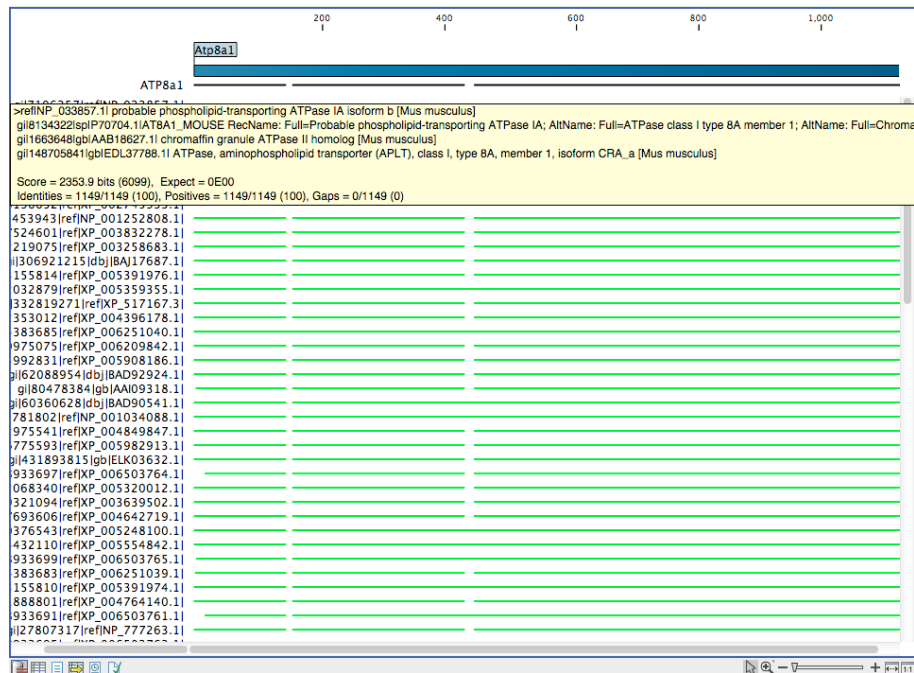
Figure 19.1: *Display of the output of a BLAST search. At the top is there a graphical representation of BLAST hits with tool-tips showing additional information on individual hits. Below is a tabular form of the BLAST results.*

19.1 Running BLAST searches

With the *CLC Main 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 you can 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 popular, and often very large, BLAST databases without having to download them to your own computer. The advantages of running BLAST on your own computer include that you can use your own sequence collections as blast databases, and that running big batch BLAST jobs can be faster and more reliable when done locally.

19.1.1 BLAST at NCBI

When running a BLAST search at the NCBI, the Workbench sends the sequences you select to the NCBI's BLAST servers. When the results are ready, they will be automatically downloaded and displayed in the Workbench. When you enter a large number of sequences for searching with BLAST, the Workbench automatically splits the sequences up into smaller subsets and sends

one subset at the time to NCBI. This is to avoid exceeding any internal limits the NCBI places on the number of sequences that can be submitted to them for BLAST searching. The size of the subset created in the CLC software depends both on the number and size of the sequences.

To start a BLAST job to search your sequences against databases held at the NCBI:

> **Toolbox | BLAST (**📚**)| BLAST at NCBI (**🔴**)**

Alternatively, use the keyboard shortcut: Ctrl+Shift+B for Windows and ⌘ +Shift+B on Mac OS.

This opens the dialog seen in figure 19.2
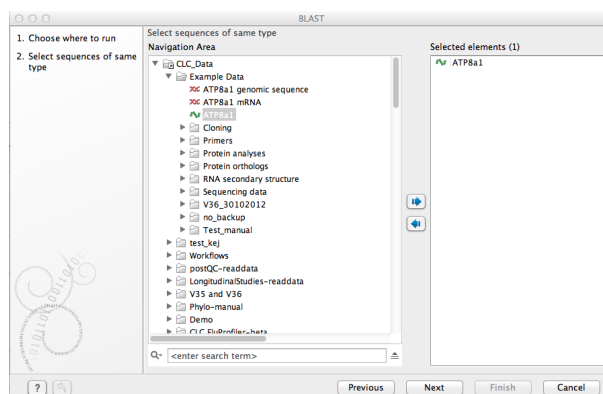
Figure 19.2: *Choose one or more sequences to conduct a BLAST search with.*

Select one or more sequences of the same type (either DNA or protein) and click **Next**.

In this dialog, you choose which type of BLAST search to conduct, and which database to search against. See figure 19.3. The databases at the NCBI listed in the dropdown box will correspond to the query sequence type you have, DNA or protein, and the type of blast search you can chose among to run. A complete list of these databases can be found in Appendix C. Here you can also read how to add additional databases available the NCBI to the list provided in the dropdown menu.
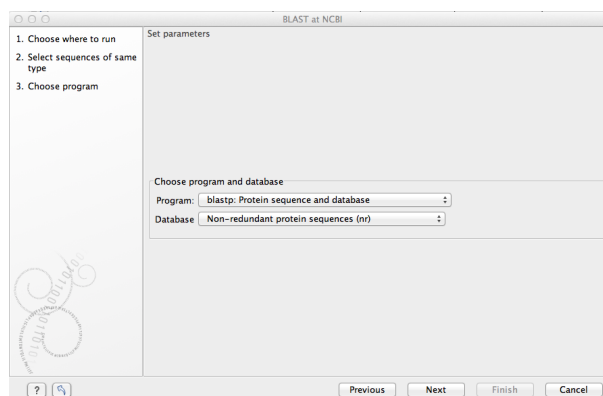
Figure 19.3: *Choose a BLAST Program and a database for the search.*

BLAST programs for DNA query sequences:

- **blastn: DNA sequence against a DNA database.** Searches for DNA sequences with homologous regions to your nucleotide query sequence.

- **blastx: Translated DNA sequence against a Protein database.** Automatic translation of

your DNA query sequence in six frames; these translated sequences are then used to search a protein database.

- **tblastx: Translated DNA sequence against a Translated DNA database.** Automatic translation of your DNA query sequence and the DNA database, in six frames. The resulting peptide query sequences are used to search the resulting peptide database. Note that this type of search is computationally intensive.

BLAST programs for protein query sequences:

- **blastp: Protein sequence against Protein database.** Used to look for peptide sequences with homologous regions to your peptide query sequence.

- **tblastn: Protein sequence against Translated DNA database.** Peptide query sequences are searched against an automatically translated, in six frames, DNA database.

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**.

This window, see figure 19.4, allows you to choose parameters to tune your BLAST search, to meet your requirements.
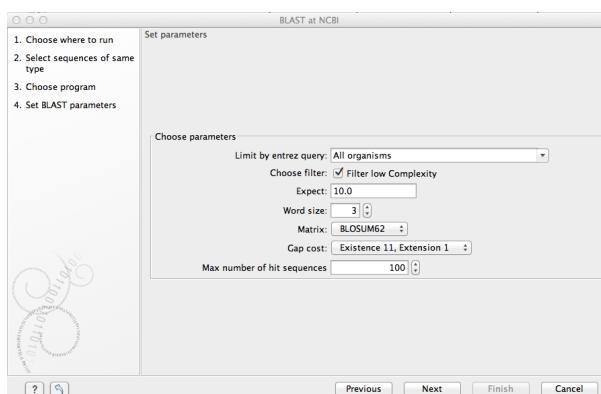
Figure 19.4: *Parameters that can be set before submitting a BLAST search.*

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 setting is particularly useful when working with organisms or organelles that have a genetic code different 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 entries in the BLAST databases. Any terms can be entered that would normally be allowed in an Entrez search session. More information about Entrez queries can be found at http://www.ncbi.nlm.nih.gov/books/NBK3837/#EntrezHelp.Entrez_Searching_Options. The syntax described there is the same as would be

accepted in the CLC interface. Some commonly used Entrez queries are pre-entered and can be chosen in the drop down menu.

- **Choose filter.** You can choose to apply **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.

- **Expect.** The threshold for reporting matches against database sequences: the default value is 10, meaning that under the circumstances of this search, 10 matches are expected to be found merely by chance according to the stochastic model of Karlin and Altschul (1990). Details of how E-values are calculated can be found at the NCBI: `http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html` If the E-value 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 results in more matches being reported, but many may just matching by chance, not due to any biological similarity. Values of E less than one can be entered as decimals, or in scientific notation. For example, 0.001, 1e-3 and 10e-4 would be equivalent and acceptable values.

- **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.

- **Match/mismatch** 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.

- **Max number of hit sequences.** The maximum number of database sequences, where BLAST found matches to your query sequence, to be included in the BLAST report.

The parameters you choose will affect how long BLAST takes to run. A search of a small database, requesting only hits that meet stringent criteria will generally be quite quick. Searching large databases, or allowing for very remote matches, will of course take longer.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

19.1.2   BLAST a partial sequence against NCBI

You can search a database using only a part of a sequence directly from the sequence view:

> **select the sequence region to send to BLAST | right-click the selection | BLAST Selection Against NCBI ( )**

This will go directly to the dialog shown in figure 19.3 and the rest of the options are the same as when performing a BLAST search with a full sequence.

19.1.3   BLAST against local data

Running BLAST searches on your local machine can have several advantages over running the searches remotely at the NCBI:

- It can be faster.

- It does not rely on having a stable internet connection.

- It does not depend on the availability of the NCBI BLAST servers.

- You can use longer query sequences.

- You use your own data sets to search against.

On a technical level, the *CLC Main Workbench* uses the NCBI's blast+ software (see `ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/`). Thus, the results of using a particular data set to search the same database, with the same search parameters, would give the same results, whether run locally or at the NCBI.

There are a number of options for what you can search against:

- You can create a database based on data already imported into your Workbench (see section 19.3.3)

- You can add pre-formatted databases (see section 19.3.1)

- You can use sequence data from the **Navigation Area** directly, without creating a database first.

To conduct a local BLAST search:

> **Toolbox | BLAST ( ) | BLAST ( )**

This opens the dialog seen in figure 19.5:

Select one or more sequences of the same type (DNA or protein) and click **Next**.

This opens the dialog seen in figure 19.6:

At the top, you can choose between different BLAST programs.

BLAST programs for DNA query sequences:

- **blastn: DNA sequence against a DNA database.** Searches for DNA sequences with homologous regions to your nucleotide query sequence.
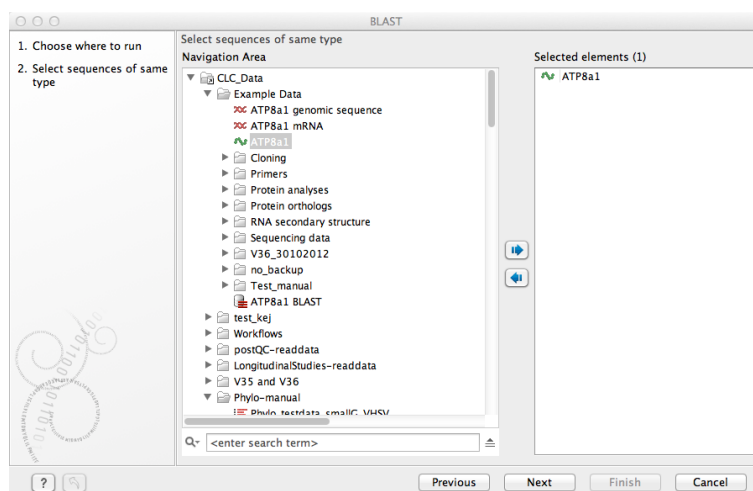
Figure 19.5: *Choose one or more sequences to conduct a BLAST search.*
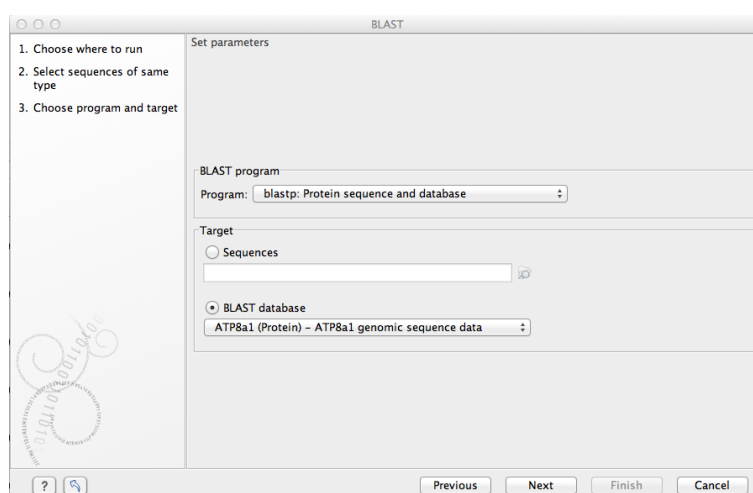
Figure 19.6: *Choose a BLAST program and a target database.*

- **blastx: Translated DNA sequence against a Protein database.** Automatic translation of your DNA query sequence in six frames; these translated sequences are then used to search a protein database.

- **tblastx: Translated DNA sequence against a Translated DNA database.** Automatic translation of your DNA query sequence and the DNA database, in six frames. The resulting peptide query sequences are used to search the resulting peptide database. Note that this type of search is computationally intensive.

BLAST programs for protein query sequences:

- **blastp: Protein sequence against Protein database.** Used to look for peptide sequences with homologous regions to your peptide query sequence.

- **tblastn: Protein sequence against Translated DNA database.** Peptide query sequences are searched against an automatically translated, in six frames, DNA database.

In cases where you have selected blastx or tblastx to conduct a search, you will get the option of selecting a translation table for the genetic code. The standard genetic code is set as default.

This setting is particularly useful when working with organisms or organelles that have a genetic code that differs from the standard genetic code.

If you search against the **Protein Data Bank** database and homologous sequences are found to the query sequence, these can be downloaded and opened with the **3D Molecule Viewer** (see section 21.1.3).

Click **Next**.

This dialog allows you to adjust the parameters to meet the requirements of your BLAST search (figure 19.7).
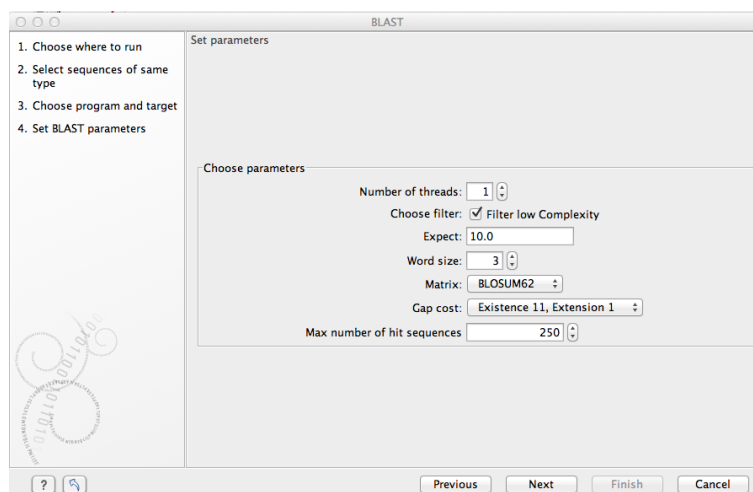
Figure 19.7: *Parameters that can be set before submitting a local BLAST search.*

- **Number of threads.** You can specify the number of threads, which should be used if your Workbench is installed on a multi-threaded system.

- **Choose filter.** You can choose to apply **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.

- **Expect.** The threshold for reporting matches against database sequences: the default value is 10, meaning that under the circumstances of this search, 10 matches are expected to be found merely by chance according to the stochastic model of Karlin and Altschul (1990). Details of how E-values are calculated can be found at the NCBI: `http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html` If the E-value 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 results in more matches being reported, but many may just matching by chance, not due to any biological similarity. Values of E less than one can be entered as decimals, or in scientific notiation. For example, 0.001, 1e-3 and 10e-4 would be equivalent and acceptable values.

- **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.

- **Match/mismatch** 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.

- **Max number of hit sequences.** The maximum number of database sequences, where BLAST found matches to your query sequence, to be included in the BLAST report.

You then specify the target database to use:

- **Sequences**. When you choose this option, you can use sequence data from the **Navigation Area** as database by clicking the **Browse and select** icon ( ). A temporary BLAST database will be created from these sequences and used for the BLAST search. It is 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 19.3.3.

- **BLAST Database**. Select a database already available in one of your designated BLAST database folders. Read more in section 19.4.

When a database or a set of sequences has been selected, click **Next**.

This opens the dialog seen in figure 19.8:

See section 19.1.1 for information about these limitations.

19.1.4 BLAST a partial sequence against a local database

You can search a database using only a part of a sequence 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 19.6 and the rest of the options are the same as when performing a BLAST search with a full sequence.
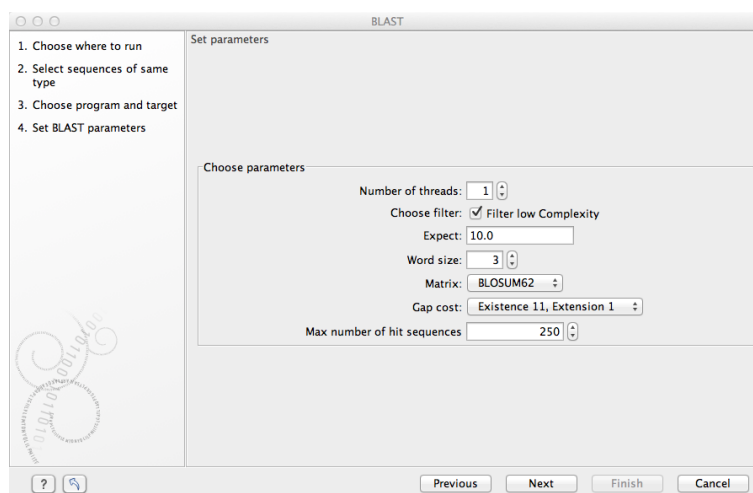
Figure 19.8: *Examples of parameters that can be set before submitting a BLAST search.*

19.2   Output from BLAST searches

The output of a BLAST search is similar whether you have chosen to run your search locally or at the NCBI.

If a **single query** sequence was used, then the results will show the hits found in that database with that single sequence. If **more than one query** sequence was used, the default view of the results is a summary table, where the description of the top match found for each query sequence and the number of matches found is reported. The summary table is described in detail in section 19.2.2.

19.2.1   Graphical overview for each query sequence

Double clicking on a given row of a tabular blast table opens a graphical overview of the blast results for a particular query sequence, as shown in figure figure 19.9. In cases where only one sequence was entered into a BLAST search, such a graphical overview is the default output.

Figure 19.9 shows an example of a BLAST result for an individual query sequence in the *CLC Main Workbench*.

Detailed descriptions of the overview BLAST table and the graphical BLAST results view are described below.

19.2.2   Overview BLAST table

In the overview BLAST table for a multi-sequence blast search, as shown in figure 19.10, there is one row for each query sequence. Each row represents the BLAST result for this query sequence.

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. Consensus sequence can be extracted by clicking the **Extract Consensus** button at the bottom. 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
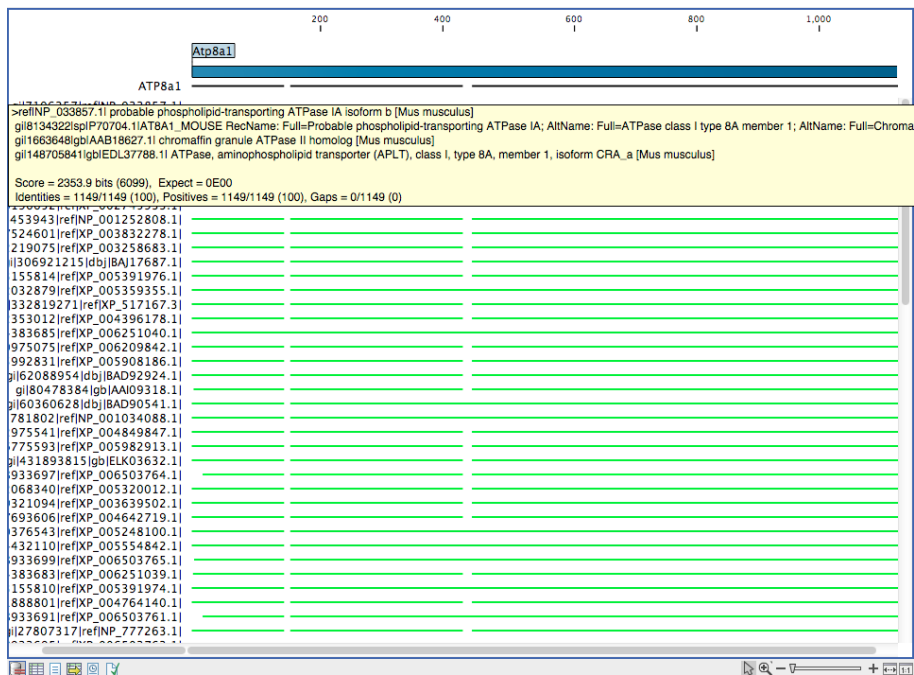
Figure 19.9: *Default display of the output of a BLAST search for one query sequence. At the top is there a graphical representation of BLAST hits with tool-tips showing additional information on individual hits.*

Figure 19.10: *An overview BLAST table summarizing the results for a number of query sequences.*

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.

- For the following list, the value of the best hit is displayed together with accession number and description of this hit, with respect to E-value, identity or positive value, hit length or bit score.

  - Lowest E-value
  - Accession (E-value)
  - Description (E-value)
  - Greatest identity %
  - Accession (identity %)
  - Description (identity %)
  - Greatest positive %
  - Accession (positive %)
  - Description (positive %)
  - Greatest hit length
  - Accession (hit length)
  - Description (hit length)
  - Greatest bit score
  - Accession (bit score)
  - Description (bit score)

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.

19.2.3   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 Settings** side panel.

- **Compactness.** 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.

- **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 adjust the coloring scale for visualisation of identity level.

The remaining View preferences for BLAST Graphics are the same as those of alignments. See section 17.1.

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.

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.

19.2.4   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.

The **BLAST table** view can be shown in the following way:

Click the Show BLAST Table button (⊞) at the bottom of the view

Figure 19.11 is an example of a BLAST Table.

Figure 19.11: *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.*

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.

- **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 Main Workbench* or *CLC Genomics Workbench*).

The hits can be sorted by the different columns, simply by clicking the column heading. In cases where individual rows have been selected in the table, the selected rows will still be selected after sorting the data.

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 19.2.3 so that selecting a hit in the table will make a selection on the corresponding sequence in the graphical view.

19.3   Local BLAST databases

BLAST databases on your local system can be made available for searches via your *CLC Main Workbench*, (section 19.3.1). To make adding databases even easier, you can download pre-formatted BLAST databases from the NCBI from within your *CLC Main Workbench*, (section 19.3.2). You can also easily create your own local blast databases from sequences within your *CLC Main Workbench*, (section 19.3.3).

19.3.1   Make pre-formatted BLAST databases available

To use databases that have been downloaded or created outside the Workbench, you can either

- Put the database files in one of the locations defined in the BLAST database manager (see section 19.4)

- Add the location where your BLAST databases are stored using the BLAST database manager (see section 19.4). See figure 19.15.

19.3.2   Download NCBI pre-formatted BLAST databases

Many popular pre-formatted databases are available for download from the NCBI. You can download any of the databases available from the list at `ftp://ftp.ncbi.nlm.nih.gov/blast/db/` from within your *CLC Main Workbench*.

You must be connected to the internet to use this tool.

If you choose:

> or   **Toolbox** | **BLAST (**📚**)** | **Download BLAST Databases (**🌐**)**

a window like the one in figure 19.12 pops up showing you the list of databases available for download.
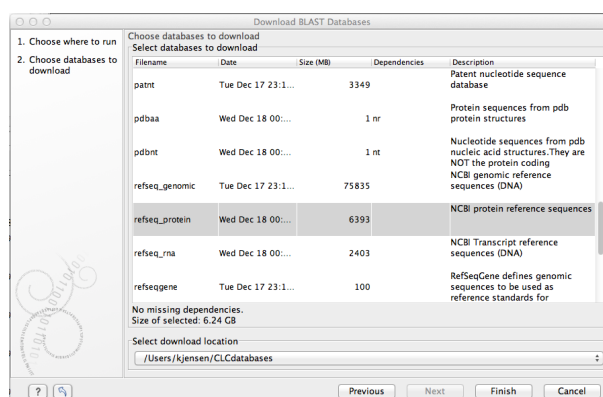
Figure 19.12: *Choose from pre-formatted BLAST databases at the NCBI available for download.*

In this window, you can see the names of the databases, the date they were made available for download on the NCBI site, the size of the files associated with that database, and a brief description of each database. You can also see whether the database has any dependencies. This aspect is described below.

You can also specify which of your database locations you would like to store the files in. Please see the **Manage BLAST Databases** section for more on this (section 19.4).

There are two very important things to note if you wish to take advantage of this tool.

- Many of the databases listed are very large. Please make sure you have space for them. If you are working on a shared system, we recommend you discuss your plans with your system administrator and fellow users.

- Some of the databases listed are dependent on others. This will be listed in the **Dependencies** column of the **Download BLAST Databases** window. This means that while the database your are interested in may seem very small, it may require that you also download a very big database on which it depends.

An example of the second item above is *Swissprot*. To download a database from the NCBI that would allow you to search just Swissprot entries, you need to download the whole *nr* database in addition to the entry for Swissprot.

19.3.3   Create local BLAST databases

In the *CLC Main Workbench* you can create a local database that you can use for local BLAST searches. You can specify a location on your computer to save the BLAST database files to. The Workbench will list the BLAST databases found in these locations when you set up a local BLAST search (see section 19.1.3).

DNA, RNA, and protein sequences located in the **Navigation Area** can be used to create BLAST databases from. Any given BLAST database can only include one molecule type. If you wish to use a pre-formatted BLAST database instead, see section 19.3.1.

To create a BLAST database, go to:

> **Toolbox** | **BLAST (📚)** | **Create BLAST Database (📗)**

This opens the dialog seen in figure 19.13.
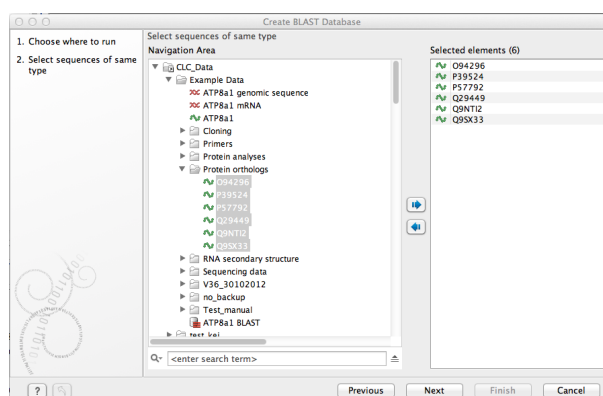
Figure 19.13: *Add sequences for the BLAST database.*

Select sequences or sequence lists you wish to include in your database and click **Next**.

In the next dialog, shown in figure 19.14, you provide the following information:

- **Name.** The name of the BLAST database. This name will be used when running BLAST searches and also as the base file name for the BLAST database files.

- **Description.** You can add more details to describe the contents of the database.

- **Location.** You can select the location to save the BLAST database files to. You can add or change the locations in this list using the **Manage BLAST Databases** tool, see section 19.4.
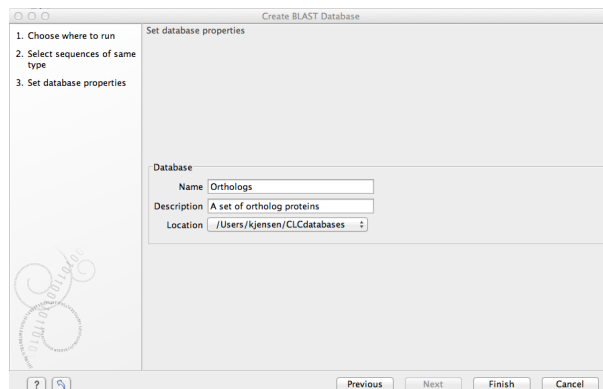
Figure 19.14: *Providing a name and description for the database, and the location to save the files to.*

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 19.4, and when running local BLAST (see section 19.1.3).

19.4   Manage BLAST databases

The BLAST databases available as targets for running local BLAST searches (see section 19.1.3) can be managed through the Manage BLAST Databases dialog (see figure 19.15):

> **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 within the Workbench using the **Create BLAST Database tool**, see section 19.3.3, or they can be pre-formatted BLAST databases.

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 have added new databases that are not listed, you can press **Refresh Locations** to clear the cache and search the database locations again.

Note: The BLAST database location and all folders in its path should **not** have any spaces in their names on Linux or Mac systems.

By default a BLAST database location will be added under your home area in a folder called CLCdatabases. This folder is scanned recursively, through all subfolders, to look for valid databases. All other folder locations are scanned only 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.
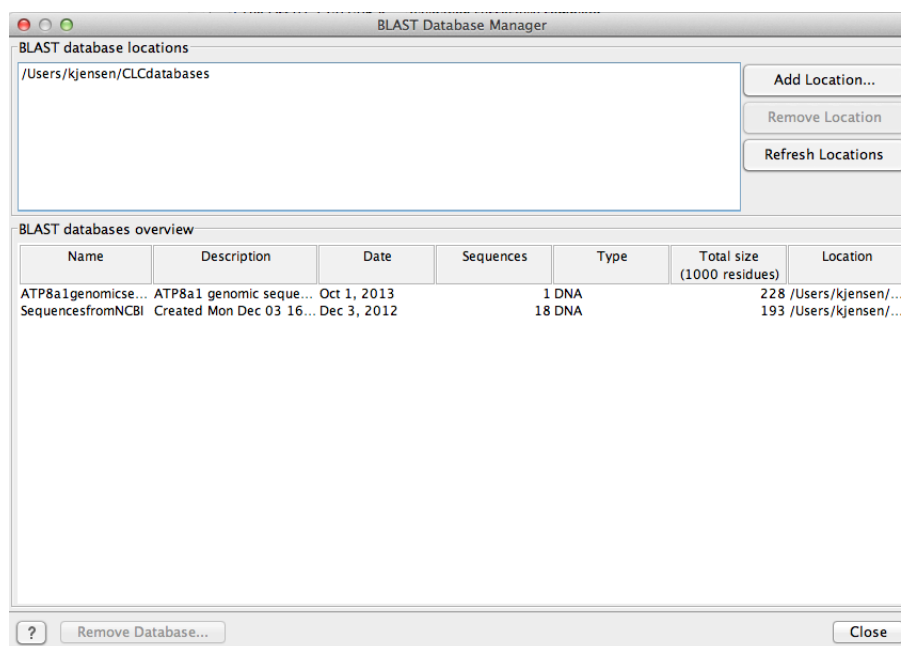
Figure 19.15: *Overview of available BLAST databases.*

- **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.

19.4.1   Migrating from a previous version of the Workbench

In versions released before 2011, 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 19.4). 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 Main Workbench 7.0* is the same as the default folder specified for saving BLAST databases in the old version.

19.5   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.

19.5.1   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.

19.5.2   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.

19.5.3   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 19.16 for an illustration of words in a protein sequence.
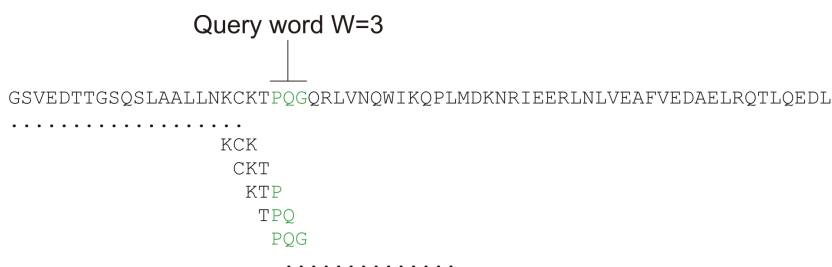
Figure 19.16: *Generation of exact BLAST words with a word size of W=3.*

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 19.16). 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 19.17). 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.
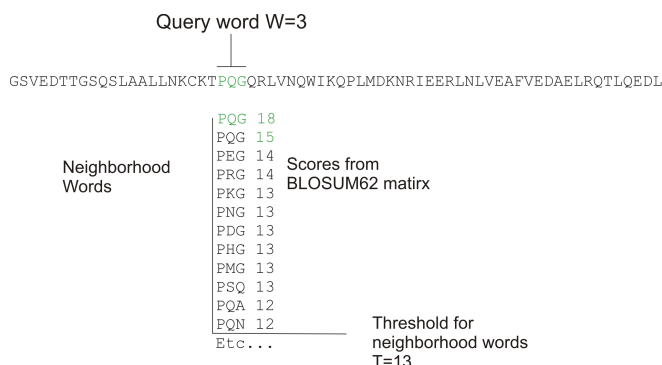
Figure 19.17: *Neighborhood BLAST words based on the BLOSUM62 matrix. Only words where the threshold* T *exceeds 13 are included in the initial seeding.*

After initial finding of words (seeding), the BLAST algorithm will extend the (only 3 residues long) alignment in both directions (see figure 19.18). 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 19.18: *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 19.19.
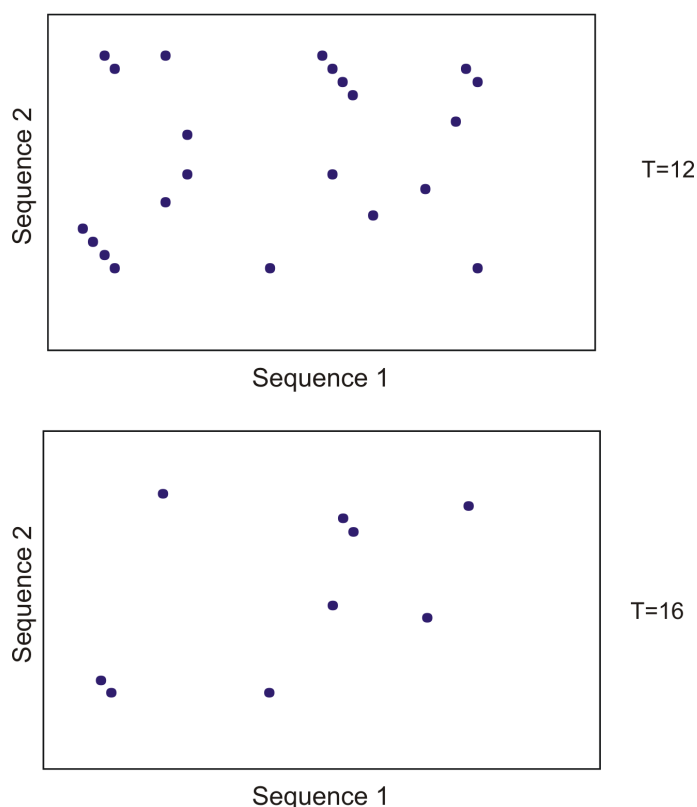
Figure 19.19: *Each dot represents a word match. Increasing the threshold of* T *limits the search space significantly.*

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.

19.5.4   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:

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.

19.5.5   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.

19.5.6   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.
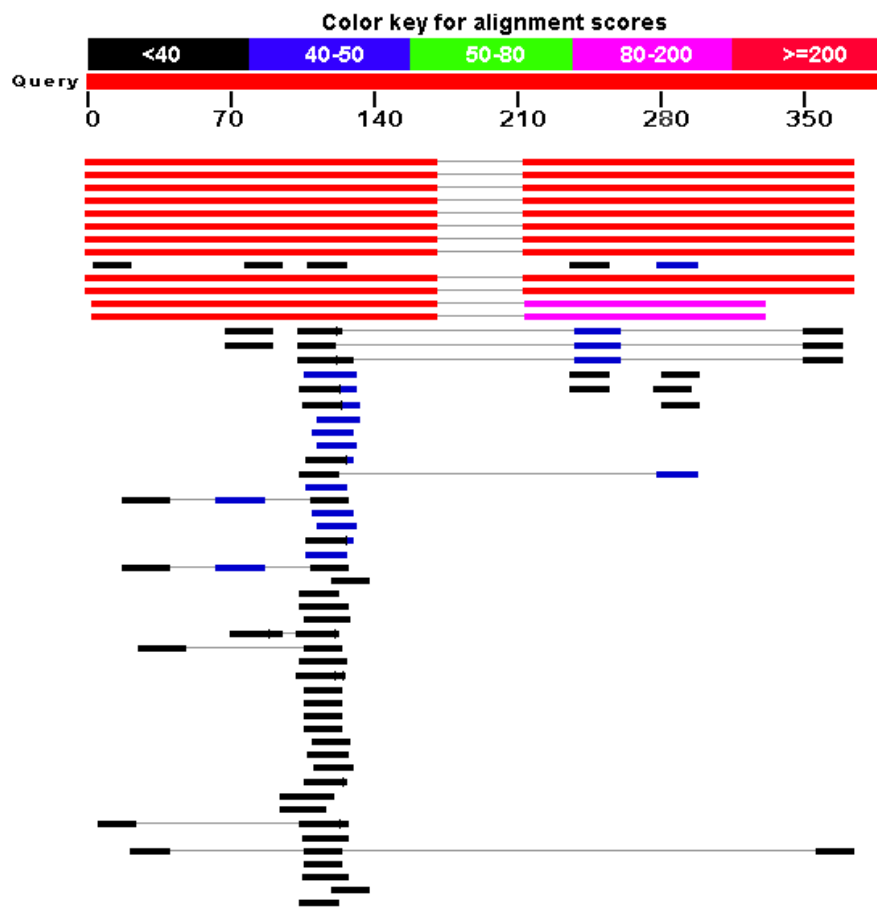
Figure 19.20: *BLAST graphical view. A simple graphical overview of the hits found aligned to the query sequence. The alignments are color coded ranging from black to red as indicated in the color label at the top.*

The graphical output (shown in figure 19.20) gives a quick overview of the query sequence and the resulting hit sequences. The hits are colored according to the obtained alignment scores.

Figure 19.21: *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.*

The table view (shown in figure 19.21) 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

Figure 19.22: *Alignment view of BLAST results. Individual alignments are represented together with BLAST scores and more.*

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.

19.5.7   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 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 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 19.23). It is also much easier to batch download a selection of hit sequences for further inspection.

Figure 19.23: *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.*

19.5.8   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.

19.5.9   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 20

3D Molecule Viewer

Contents

Proteins are amino acid polymers that are involved in all aspects of cellular function. The structure of a protein is defined by its particular amino acid sequence, with the amino acid sequence being referred to as the primary protein structure. The amino acids fold up in local structural elements; helices and sheets, also called the secondary structure of the protein. These structural elements are then packed into globular folds, known as the tertiary structure or the three dimensional structure.

In order to understand protein function it is often valuable to see the three dimensional structure of the protein. This is possible when the structure of the protein has been resolved and published. Structure files are usually deposited in the Protein Data Bank (PDB) http://www.rcsb.org/, where the publicly available protein structure files can be searched and downloaded. The vast majority of the protein structures have been determined by X-ray crystallography (88%) while the rest of the structures predominantly have been obtained by Nuclear Magnetic Resonance techniques.

In addition to protein structures, the PDB entries also contain structural information about molecules that interact with the protein, such as nucleic acids, ligands, cofactors, and water.

There are also entries, which contain nucleic acids and no protein structure. The **3D Molecule Viewer** in the *CLC Main Workbench* is an integrated viewer of such structure files.

The **3D Molecule Viewer** offers a range of tools for inspection and visualization of the molecular structures in the Molecule Project:

- Automatic sorting of molecules into categories; Proteins, Nucleic acids, Ligands, Cofactors, Water molecules

- Hide/unhide individual molecules from the view

- Four different atom-based molecule visualizations

- Backbone visualization for proteins and nucleic acids

- Molecular surface visualization

- Selection of different color schemes for each molecule visualization

- Customized visualization for user selections

- Browse amino acids and nucleic acids from sequence editors started from within the 3D Molecule Viewer

20.1   Importing molecule structure files

The supported file format for three dimensional protein structures in the **3D Molecule Viewer**, is the Protein Data Bank (PDB) format, which upon import is converted to a CLC Molecule Project. PDB files can be imported to a Molecule Project in three different ways:

- **From the Protein Data Bank** (21.1.1)

- **From your own file system** (21.1.2)

- **Using BLAST search against the PDB database** (21.1.3)

20.1.1   From the Protein Data Bank

Molecule structures can be imported in the workbench from the Protein Data Bank using the "Download" function:

> **Toolbar** | **Download** ( ) | **Search for PDB structures at NCBI** ( )

Type the molecule name or accession number into the search field and click on the "Start search" button (as shown in figure 21.1). The search hits will appear in the table below the search field.

Select the molecule structure of interest and click on the button labeled "Download and Open" (see figure 21.1) or double click on the relevant row in the table to open the protein structure.

Pressing the "Download and Save" button will save the molecule structure at a user defined destination in the **Navigation Area**.

The button "Open at NCBI" links directly to the structure summary page at NCBI. Clicking this button will open individual NCBI pages describing each of the selected molecule structures.

Figure 20.1: *Download protein structure from the Protein Data Bank. It is possible to open a structure file directly from the output of the search by clicking the "Download and Open" button or by double clicking directly on the relevant row.*

20.1.2 From your own file system

A PDB file can also be imported from your own file system using the standard import function:

> **Toolbar** | **Import** (📥) | **Standard Import** (📥)

In the Import dialog, select the structure(s) of interest from a data location and tick "Automatic import" (figure 21.2). Specify where to save the imported PDB file and click **Finish**.

Double clicking on the imported file in the **Navigation Area** will open the structure as a **Molecule Project** in the **View Area** of the *CLC Main Workbench*. Another option is to drag the PDB file from the **Navigation Area** to the **View Area**. This will automatically open the protein structure as a **Molecule Project**.

Figure 20.2: *A PDB file can be imported using the "Standard Import" function.*

20.1.3   BLAST search against the PDB database

It is also possible to make a BLAST search against the PDB database.

Toolbox | BLAST ( )| BLAST at NCBI ( )

After selecting where to run the analysis, specify which input sequence to use for the BLAST search in the "BLAST at NCBI" dialog "Select sequences of same type". More than one sequence can be selected at the time as long as the sequences are of the same type (figure 21.3).
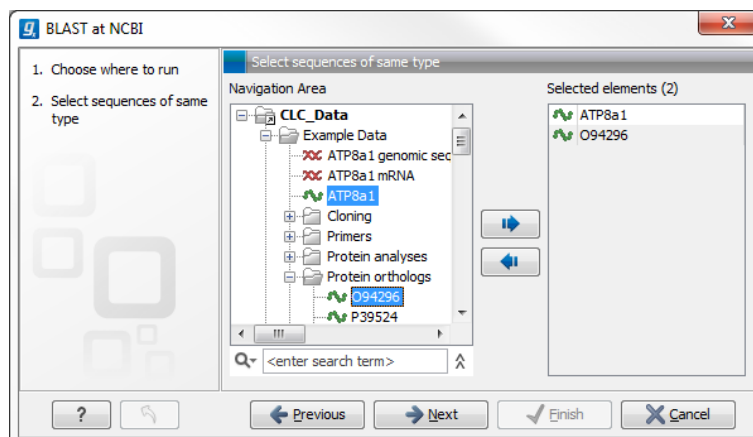
Figure 20.3: *Select the input sequence of interest. In this example a protein sequence for ATPase class I type 8A member 1 and an ATPase ortholog from S. pombe have been selected.*

Click **Next** and choose program and database (figure 21.4). When a protein sequence has been used as input, select "Program: blastp: Protein sequence and database" and "Database: Protein Data Bank proteins (pdb)".

It is also possible to use mRNA and genomic sequences as input. In such cases the program "blastx: Translated DNA sequence and protein database" should be used.
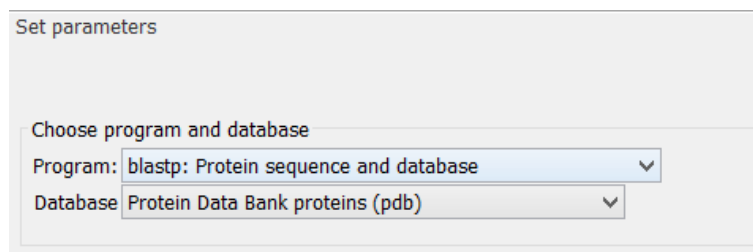
Figure 20.4: *Select database and program.*

Please refer to section 19.1.1 for further description of the individual parameters in the wizard steps.

When you click on the button labeled **Finish**, a BLAST output is generated that shows local sequence alignments between your input sequence and a list of matching proteins with known structures available.

Note! The BLAST at NCBI search can take up to several minutes, especially when mRNA and genomic sequences are used as input.

Switch to the "BLAST Table" editor view to select the desired entry (figure 21.5). If you have performed a multi BLAST, to get access to the "BLAST Table" view, you must first double click

on each row to open the entries individually.

In this view four different options are available:

- **Download and Open** The sequence that has been selected in the table is downloaded and opened in the **View Area**.

- **Download and Save** The sequence that has been selected in the table is downloaded and saved in the **Navigation Area**.

- **Open at NCBI** The protein sequence that has been selected in the table is opened at NCBI.

- **Open Structure** Opens the structure in a **Molecule Project** in the **View Area**.

Figure 20.5: *Top: The output from "BLAST at NCBI". Bottom: The "BLAST table". One of the protein sequences has been selected. This activates the four buttons under the table. Note that the table and the BLAST Graphics are linked, this means that when a sequence is selected in the table, the same sequence will be highlighted in the BLAST Graphics view.*

20.1.4 Import issues

When opening an imported molecule file for the first time, a notification is briefly shown in the lower left corner of the **Molecule Project** editor, with information of the number of issues

encountered during import of the file. The issues are categorized and listed in a table view in the Issues editor. The Issues editor can be opened by selecting **Show | Issue Editor** from the menu appearing when right-clicking in an empty space in the 3D view (figure 21.6).

Alternatively, the editor can be accessed from the lower left corner of the view, where buttons are shown for each available editor. If you hold down the Ctrl key while clicking on the editor icon, it will be shown in a split view together with the 3D view. The issues table is linked with the molecules in the 3D view, such that selecting an entry in the table will select the implicated atoms in the view and zoom to put them into the center of the 3D view.

Figure 20.6: *At the bottom of the Molecule Project it is possible to switch to the "Show Issues Editor" view by clicking on the "table-with-exclamation-mark" icon.*

20.2 Viewing molecular structures in 3D

An example of a 3D structure that has been opened as a **Molecule Project** is shown in figure 21.7.

20.2.1 Moving and rotating

The molecules can be rotated by holding down the left mouse button while moving the mouse. The right mouse button can be used to move the view.

Zooming can be done with the scroll-wheel or by holding down both left and right buttons while moving the mouse up and down.

All molecules in the **Molecule Project** are listed in categories in the **Project Tree**. The individual molecules or whole categories can be hidden from the view by un-cheking the boxes next to them.

It is possible to bring a particular molecule or a category of molecules into focus by selecting
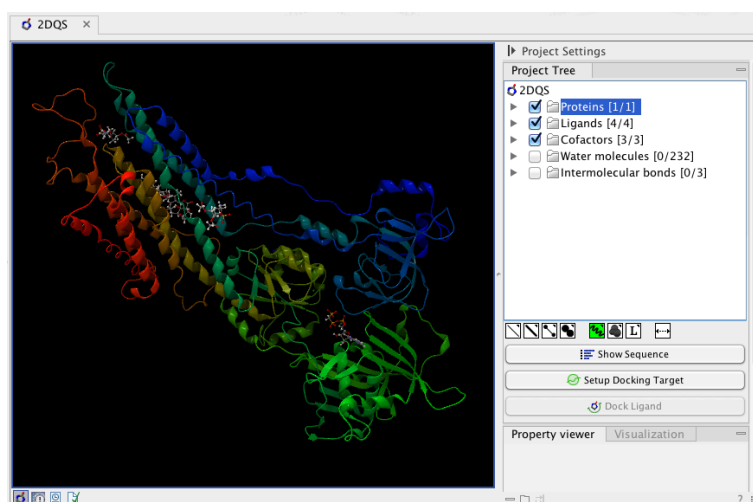
Figure 20.7: *3D view of a calcium ATPase. All molecules in the PDB file are shown in the Molecule Project. The Project Tree in the right side of the window lists the involved molecules.*

the molecule or category of interest in the **Project Tree** view and double-click on the molecule or category of interest. Another option is to use the zoom-to-fit button (⟵⋯⟶) at the bottom of the **Project Tree** view.

20.3   Customizing the visualization

The molecular visualization of all molecules in the Molecule Project can be customized using different visualization styles. The styles can be applied to one molecule at a time, or to a whole category (or a mixture), by selecting the name of either the molecule or the category. Holding down the Ctrl (Cmd on Mac) or shift key while clicking the entry names in the **Project Tree** will select multiple molecules/categories.

Quick-style buttons below the **Project Tree** view give access to the molecule visualization styles, while context menus on the buttons (accessible via right-click or left-click-hold) give access to the color schemes available for the visualization styles. Visualization styles and color schemes are also available from context menus directly on the selected entries in the **Project Tree**.

Note! Whenever you wish to change the visualization styles by right-clicking the entries in the **Project Tree**, please be aware that you must first click on the entry of interest, and ensure it is highlighted in blue, before right-clicking on the entry.

20.3.1   Visualization styles and colors

Wireframe, Stick, Ball and stick, Space-filling/CPK

(◥) (◥) (◥) (🏀)

Four different ways of visualizing molecules by showing all atoms are provided: Wireframe, Stick, Ball and stick, Space-filling/CPK.

The visualizations are mutually exclusive meaning that only one style can be applied at a time for each selected group of atoms.

Four color schemes are available and can be accessed via right-clicking on the visualization style

icons:

- **Color by Element** Classic CPK coloring based on atom type (e.g. oxygen red, carbon gray, hydrogen white, nitrogen blue, sulfur yellow).

- **Color by Temperature** This is based on the b-factors in the PDB file and a color scale going from blue (0) over white (50) to red (100). The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **Color Different Objects** Each molecule is assigned its own random color.

- **Custom Color** The user selects molecule colors from a palette.

Backbone

(🐛)

For the molecules in the Proteins and Nucleic Acids categories, the backbone structure can be visualized in a schematic rendering, highlighting the secondary structure elements for proteins and matching base pairs for nucleic acids. The backbone visualization can be combined with any of the atom-level visualizations.

Five color schemes are available for backbone structures:

- **Color by Type** For proteins, beta sheets are blue, helices red and loops/coil gray. For nucleic acids backbone ribbons are white while the individual nucleotides are indicated in green (T/U), red (A), yellow (G), and blue (C).

- **Color by Residue Position** Rainbow color scale going from blue over green to yellow and red, following the residue number

- **Color Different Chains** Each chain/molecule is assigned its own random color.

- **Color by Backbone Temperature** based on the b-factors for the Cα atoms (the central carbon atom in each amino acid) and a color scale going from blue (0) over white (50) to red (100). The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **Custom Color** The user selects molecule colors from a palette.

Surfaces

(🪨)

Molecular surfaces can be visualized.

Five color schemes are available for surfaces:

- **Color by Charge** Charged amino acids close to the surface will show as red (negative) or blue (positive) areas on the surface, with a color gradient that depends on the distance of the charged atom to the surface.

- **Color Different Surfaces** Each surface is assigned its own random color.

- **Color by Element** Smoothed out coloring based on the classic CPK coloring of the heteroatoms close to the surface.

- **Color by Temperature** Smoothed out coloring based on the b-factors for the atoms close to the surface. The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder. The color scale goes from blue (0) over white (50) to red (100).

- **Custom Color** The user selects molecule colors from a palette.

A surface spanning multiple molecules can be visualized by making a custom atom group that includes all atoms from the molecules (see section 21.3.1)

Labels

(**L**)

Labels can be added to the molecules in the view by selecting an entry in the Project Tree and clicking the label button at the bottom of the Project Tree view. The color of the labels can be adjusted from the context menu by right clicking on the selected entry (which must be highlighted in blue first) or on the label button in the bottom of the Project Tree view (see figure 21.8).

Figure 20.8: *The color of the labels can be adjusted in two different ways. Either directly using the label button by right clicking the button, or by right clicking on the molecule or category of interest in the Project Tree.*

- For proteins and nucleic acids, each residue is labelled with the PDB name and number.

- For ligands, each atom is labelled with the atom name as given in the input.

- For cofactors and water, one label is added with the name of the molecule.

Labels can be removed again by clicking on the label button.

Zoom to fit

(⟵⋯⟶)

The "Zoom to fit" button can be used to automatically move a region of interest into the center of the screen. This can be done by selecting a molecule or category of interest in the **Project Tree** view followed by a click on the "Zoom to fit" button (⟵⋯⟶) at the bottom of the **Project Tree**

view (figure 21.9). Double-clicking an entry in the **Project Tree** will have the same effect.

Figure 20.9: *The "Fit to screen" button can be used to bring a particular molecule or category of molecules in focus.*

Custom atom groups

In some situations it may be relevant to use a unique visualization style or color to highlight a particular set of atoms, or to visualize only a subset of atoms from a molecule. This can be done by making an atom group selection. Atoms can be selected in different ways, as listed below. When an atom group has been created, it appears as an entry in the **Project Tree** in the category "Selections". The atoms can then be hidden or shown, and the visualization changed, just as for the molecules in the **Project Tree**.

Atom groups can be deleted from the context menu, when selecting them in the Project Tree.

How to select a particular group of atoms

A group of atoms can be selected in different ways. Brown spheres will indicate which atoms are selected in the 3D view. The selection will appear as the entry "Current" in the Selections category in the **Project Tree**. To convert the current selection into an atom group, select the Current entry in the **Project Tree**, and from the right-click context menu choose 'Create Group from Selection' to make an atom group with exactly the selected atoms, or choose 'Create Group from Selection plus Context' to include whole residues or molecules in the atom group.

- **Double click to select** Click on an atom to select it. When you double click on an atom that belongs to a residue in a protein or in a nucleic acid chain, the entire residue will be selected. For small molecules, the entire molecule will be selected.

- **Adding atoms to a selection** Holding down Ctrl while picking atoms, will pile up the atoms in the selection. All atoms in a molecule or category from the Project Tree, can be added

Figure 20.10: *An atom group that has been highlighted by adding a unique visualization style.*

to the "Current" selection by choosing "Add to Selection" in the context menu. Similarly, whole molecules can be removed from the current selection via the context menu.

- **Spherical selection** Hold down the shift-key, click on an atom and drag the mouse away from the atom. Then a sphere centered on the atom will appear, and all visible atoms inside the sphere will be selected. The status bar (lower right corner) will show the radius of the sphere.

- **Show Sequence** Another option is to select protein or nucleic acid entries in the Project Tree, and click the 'Show Sequence' button found below the Project Tree. A split-view will appear with a sequence list editor for each of the sequence data types (Protein, DNA, RNA) (figure 21.11). If you go to the protein (or nucleic acid) sequence that was opened with the "Show Sequence" button and select a region in the sequence, the selected residues will show up as the "Current" selection in the 3D view and the Project Tree view. Notice that the link between the 3D view and the sequence editor is lost if either window is closed, or if the sequences are modified.

20.3.2   Project settings

A number of general settings can be adjusted from the **Side Panel**. Personal settings can be saved by clicking in the lower right corner of the **Side Panel** (⬚). This is described in detail in section 11.6.

Project Tree Actions

Just below the Project Tree, the following action is available

- **Show Sequence** Select molecules which have sequences associated (Protein, DNA, RNA) in the Project Tree, and click this button. Then, a split-view will appear with a sequence list editor for each of the sequence data types (Protein, DNA, RNA). This is described in section 21.5.
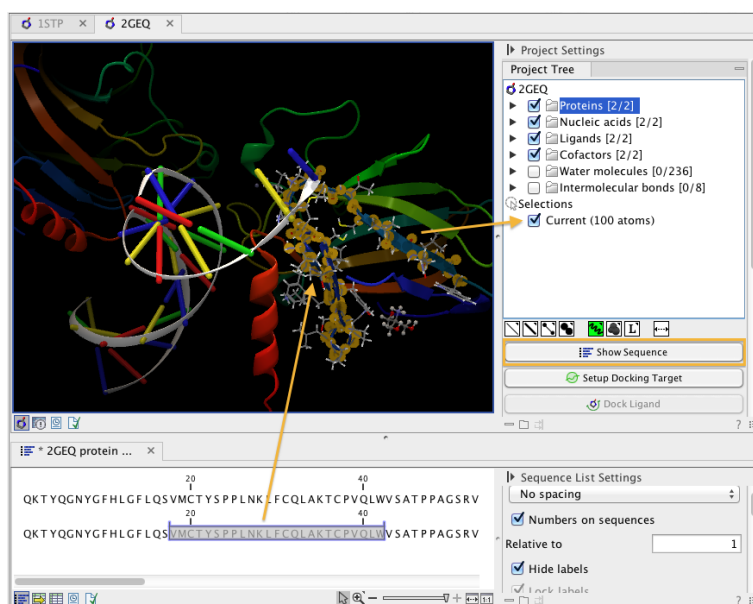
Figure 20.11: *The protein sequence in the split view is linked with the protein structure. This means that when a part of the protein sequence is selected, the same region in the protein structure will be selected.*

Property viewer

The Property viewer, found in the Side Panel, lists detailed information about the atoms that the mouse hovers over. For all atoms the following information is listed:

- **Name** The particular atom name, if given in input.

- **Element** The element type (C, N, O,...).

- **Hybridization** The atom hybridization assigned to the atom.

- **Charge** The atomic charge as given in the input file. If charges are not given in the input file, some charged chemical groups are automatically recognized and a charge assigned.

- **Molecule** The name of the molecule the atom is part of.

- **Chain** For proteins and nucleic acids, the name of the chain the atom belongs to is listed.

- **Residue** For proteins and nucleic acids, the name and number of the residue the atom belongs to is listed.

For atoms in molecules imported from a PDB file, extra information is given:

- **Temperature** Here is listed the b-factor assigned to the atom in the PDB file. The b-factor is a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **PDB index** Each atom listed in the PDB file is given an index number, which is listed in the second column of the ATOM or HETATOM entries in the PDB file.

- **Source line** The line number in the PDB text file where the atom information appears.

If an atom is selected, the Property view will be frozen with the details of the selected atom shown. If then a second atom is selected (by holding down Ctrl while clicking), the distance between the two selected atoms is shown. If a third atom is selected, the angle for the second atom selected is shown. If a fourth atom is selected, the dihedral angle measured as the angle between the planes formed by the three first and three last selected atoms is given.

Figure 20.12: *Selecting two, three, or four atoms will display the distance, angle, or dihedral angle, respectively.*

If a molecule is selected in the Project Tree, the Property view shows information about this molecule:

- **Atoms** Number of atoms in the molecule.

- **Weight** The weight of the molecule in Daltons.

Visualization settings

Under "Visualization" four options exist:

- **Hydrogens** Hydrogen atoms can be shown (Show all hydrogens), hidden (Hide all hydrogens) or partially shown (Show only polar hydrogens).

- **Fog** "Fog" is added to give a sense of depth in the view. The strength of the fog can be adjusted or it can be disabled.

- **3D projection** The view is opened up towards the viewer, with a "Perspective" 3D projection. The field of view of the perspective can be adjusted, or the perspective can be disabled by selecting an orthographic 3D projection.

- **Coloring** The background color can be selected from a color palette by clicking on the colored box.

20.4 Snapshots of the molecule visualization

To save the current view as a picture, right-click in the **View Area** and select "File" and "Export Graphics". Another way to save an image is by pressing the "Graphics" button in the Workbench toolbar ( ). Next, select the location where you wish to save the image, select file format (PNG, JPEG, or TIFF), and provide a name, if you wish to use another name than the default name.

20.5 Sequences associated with the molecules

From the Side Panel, sequences associated with the molecules in the Molecule Project can be opened as separate objects by selecting molecule entries in the Project Tree and clicking the button labeled "Show Sequence" (figure 21.13). This will generate a sequence list for each selected sequence type (protein, DNA, RNA). The sequence list can be used to select atoms in the Molecular Project as described in (section 21.3.1). The sequence list can also be saved as an independent object and used as input for sequence analysis tools.
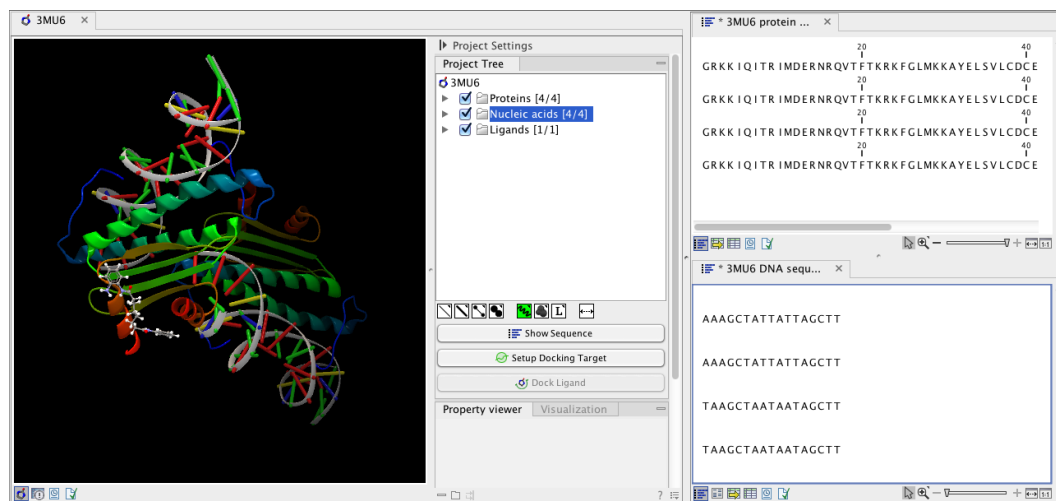
Figure 20.13: *All protein chain sequences as well as DNA sequences of interacting DNA are shown as individual sequences.*

20.6 Troubleshooting 3D graphics errors

The 3D viewer uses OpenGL graphics hardware acceleration in order to provide the best possible experience. If you experience any graphics problems with the 3D view, please make sure that the drivers for your graphics card are up-to-date.

If the problems persist after upgrading the graphics card drivers, it is possible to change to a rendering mode, which is compatible with a wider range of graphic cards. To change the graphics mode go to Edit in the menu bar, select "Preferences", Click on "View", scroll down to the bottom and find "Molecule Project 3D Editor" and uncheck the box "Use modern OpenGL rendering".

Finally, it should be noted that certain types of visualization are more demanding than others. In particular, using multiple molecular surfaces may result in slower drawing, and even result in the graphics card running out of available memory. Consider creating a single combined surface (by using a selection) instead of creating surfaces for each single object. For molecules with a large number of atoms, changing to wireframe rendering and hiding hydrogen atoms can also greatly improve drawing speed.

20.7 Updating old structure files

As the 3D Molecule Viewer has been completely redesigned, it is necessary to update old structure files. To update existing structure files, double click on the name in the **Navigation Area**. This will bring up the dialog shown in figure 21.14, which via the "Download from PDB..." button gives access to downloading the specific structure in PDB format.
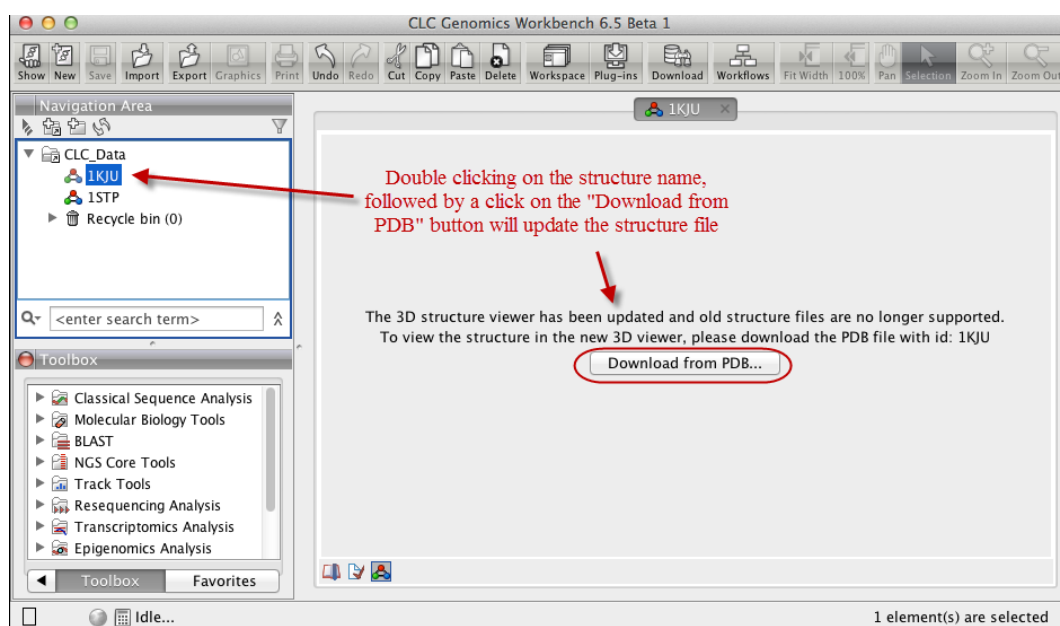
Figure 20.14: *Old structure files are not supported by the new 3D Molecule Viewer and must be updated.*

Chapter 21

Protein viewer

Contents

Proteins are amino acid polymers that are involved in all aspects of cellular function. The structure of a protein is defined by its particular amino acid sequence, with the amino acid sequence being referred to as the primary protein structure. The amino acids fold up in local structural elements; helices and sheets, also called the secondary structure of the protein. These structural elements are then packed into globular folds, known as the tertiary structure or the three dimensional structure.

In order to understand protein function it is often valuable to see the three dimensional structure of the protein. This is possible when the structure of the protein has been resolved and published. Structure files are usually deposited in the Protein Data Bank (PDB) http://www.rcsb.org/, where the publicly available protein structure files can be searched and downloaded. The vast majority of the protein structures have been determined by X-ray crystallography (88%) while the rest of the structures predominantly have been obtained by Nuclear Magnetic Resonance techniques.

In addition to protein structures, the PDB entries also contain structural information about molecules that interact with the protein, such as nucleic acids, ligands, cofactors, and water.

327

There are also entries, which contain nucleic acids and no protein structure. The **3D Molecule Viewer** in the *CLC Main Workbench* is an integrated viewer of such structure files.

The **3D Molecule Viewer** offers a range of tools for inspection and visualization of the molecular structures in the Molecule Project:

- Automatic sorting of molecules into categories; Proteins, Nucleic acids, Ligands, Cofactors, Water molecules

- Hide/unhide individual molecules from the view

- Four different atom-based molecule visualizations

- Backbone visualization for proteins and nucleic acids

- Molecular surface visualization

- Selection of different color schemes for each molecule visualization

- Customized visualization for user selections

- Browse amino acids and nucleic acids from sequence editors started from within the 3D Molecule Viewer

21.1   Importing molecule structure files

The supported file format for three dimensional protein structures in the **3D Molecule Viewer**, is the Protein Data Bank (PDB) format, which upon import is converted to a CLC Molecule Project. PDB files can be imported to a Molecule Project in three different ways:

- **From the Protein Data Bank** (21.1.1)

- **From your own file system** (21.1.2)

- **Using BLAST search against the PDB database** (21.1.3)

21.1.1   From the Protein Data Bank

Molecule structures can be imported in the workbench from the Protein Data Bank using the "Download" function:

Toolbar | Download ( ) | Search for PDB structures at NCBI ( )

Type the molecule name or accession number into the search field and click on the "Start search" button (as shown in figure 21.1). The search hits will appear in the table below the search field.

Select the molecule structure of interest and click on the button labeled "Download and Open" (see figure 21.1) or double click on the relevant row in the table to open the protein structure.

Pressing the "Download and Save" button will save the molecule structure at a user defined destination in the **Navigation Area**.

The button "Open at NCBI" links directly to the structure summary page at NCBI. Clicking this button will open individual NCBI pages describing each of the selected molecule structures.

Figure 21.1: *Download protein structure from the Protein Data Bank. It is possible to open a structure file directly from the output of the search by clicking the "Download and Open" button or by double clicking directly on the relevant row.*

21.1.2 From your own file system

A PDB file can also be imported from your own file system using the standard import function:

> **Toolbar** | **Import** (📥) | **Standard Import** (📥)

In the Import dialog, select the structure(s) of interest from a data location and tick "Automatic import" (figure 21.2). Specify where to save the imported PDB file and click **Finish**.

Double clicking on the imported file in the **Navigation Area** will open the structure as a **Molecule Project** in the **View Area** of the *CLC Main Workbench*. Another option is to drag the PDB file from the **Navigation Area** to the **View Area**. This will automatically open the protein structure as a **Molecule Project**.

Figure 21.2: *A PDB file can be imported using the "Standard Import" function.*

21.1.3 BLAST search against the PDB database

It is also possible to make a BLAST search against the PDB database.

Toolbox | BLAST ( )| BLAST at NCBI ( )

After selecting where to run the analysis, specify which input sequence to use for the BLAST search in the "BLAST at NCBI" dialog "Select sequences of same type". More than one sequence can be selected at the time as long as the sequences are of the same type (figure 21.3).
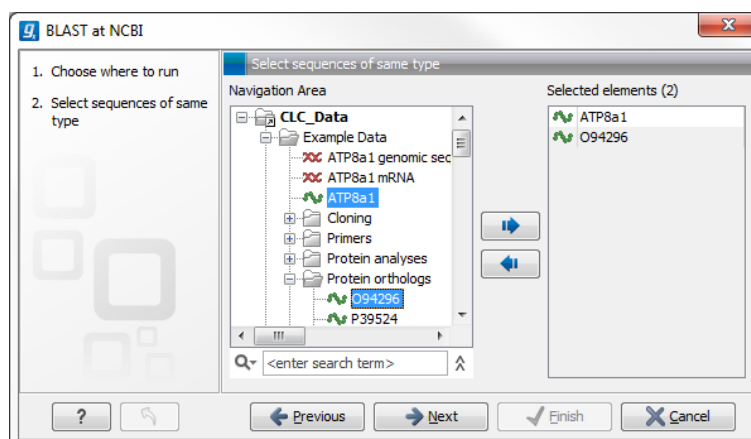
Figure 21.3: *Select the input sequence of interest. In this example a protein sequence for ATPase class I type 8A member 1 and an ATPase ortholog from S. pombe have been selected.*

Click **Next** and choose program and database (figure 21.4). When a protein sequence has been used as input, select "Program: blastp: Protein sequence and database" and "Database: Protein Data Bank proteins (pdb)".

It is also possible to use mRNA and genomic sequences as input. In such cases the program "blastx: Translated DNA sequence and protein database" should be used.
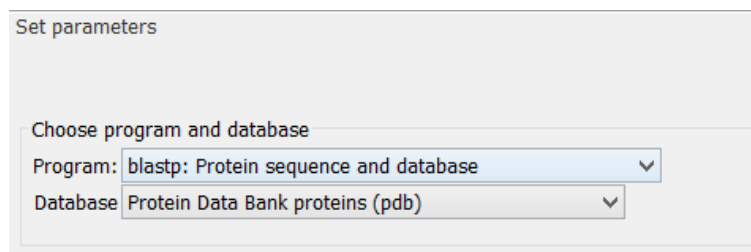
Figure 21.4: *Select database and program.*

Please refer to section 19.1.1 for further description of the individual parameters in the wizard steps.

When you click on the button labeled **Finish**, a BLAST output is generated that shows local sequence alignments between your input sequence and a list of matching proteins with known structures available.

Note! The BLAST at NCBI search can take up to several minutes, especially when mRNA and genomic sequences are used as input.

Switch to the "BLAST Table" editor view to select the desired entry (figure 21.5). If you have performed a multi BLAST, to get access to the "BLAST Table" view, you must first double click

on each row to open the entries individually.

In this view four different options are available:

- **Download and Open** The sequence that has been selected in the table is downloaded and opened in the **View Area**.

- **Download and Save** The sequence that has been selected in the table is downloaded and saved in the **Navigation Area**.

- **Open at NCBI** The protein sequence that has been selected in the table is opened at NCBI.

- **Open Structure** Opens the structure in a **Molecule Project** in the **View Area**.

Figure 21.5: *Top: The output from "BLAST at NCBI". Bottom: The "BLAST table". One of the protein sequences has been selected. This activates the four buttons under the table. Note that the table and the BLAST Graphics are linked, this means that when a sequence is selected in the table, the same sequence will be highlighted in the BLAST Graphics view.*

21.1.4   Import issues

When opening an imported molecule file for the first time, a notification is briefly shown in the lower left corner of the **Molecule Project** editor, with information of the number of issues

encountered during import of the file. The issues are categorized and listed in a table view in the Issues editor. The Issues editor can be opened by selecting **Show | Issue Editor** from the menu appearing when right-clicking in an empty space in the 3D view (figure 21.6).

Alternatively, the editor can be accessed from the lower left corner of the view, where buttons are shown for each available editor. If you hold down the Ctrl key while clicking on the editor icon, it will be shown in a split view together with the 3D view. The issues table is linked with the molecules in the 3D view, such that selecting an entry in the table will select the implicated atoms in the view and zoom to put them into the center of the 3D view.

Figure 21.6: *At the bottom of the Molecule Project it is possible to switch to the "Show Issues Editor" view by clicking on the "table-with-exclamation-mark" icon.*

21.2   Viewing molecular structures in 3D

An example of a 3D structure that has been opened as a **Molecule Project** is shown in figure 21.7.

21.2.1   Moving and rotating

The molecules can be rotated by holding down the left mouse button while moving the mouse. The right mouse button can be used to move the view.

Zooming can be done with the scroll-wheel or by holding down both left and right buttons while moving the mouse up and down.

All molecules in the **Molecule Project** are listed in categories in the **Project Tree**. The individual molecules or whole categories can be hidden from the view by un-cheking the boxes next to them.

It is possible to bring a particular molecule or a category of molecules into focus by selecting
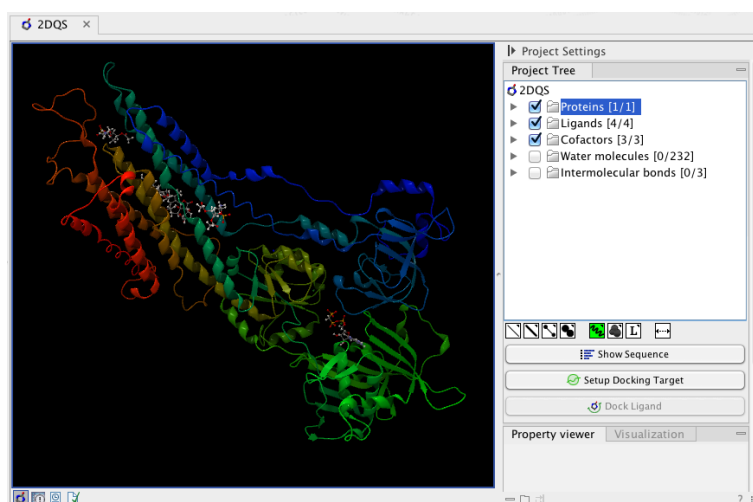
Figure 21.7: *3D view of a calcium ATPase. All molecules in the PDB file are shown in the Molecule Project. The Project Tree in the right side of the window lists the involved molecules.*

the molecule or category of interest in the **Project Tree** view and double-click on the molecule or category of interest. Another option is to use the zoom-to-fit button (⟵⋯➔) at the bottom of the **Project Tree** view.

21.3 Customizing the visualization

The molecular visualization of all molecules in the Molecule Project can be customized using different visualization styles. The styles can be applied to one molecule at a time, or to a whole category (or a mixture), by selecting the name of either the molecule or the category. Holding down the Ctrl (Cmd on Mac) or shift key while clicking the entry names in the **Project Tree** will select multiple molecules/categories.

Quick-style buttons below the **Project Tree** view give access to the molecule visualization styles, while context menus on the buttons (accessible via right-click or left-click-hold) give access to the color schemes available for the visualization styles. Visualization styles and color schemes are also available from context menus directly on the selected entries in the **Project Tree**.

Note! Whenever you wish to change the visualization styles by right-clicking the entries in the **Project Tree**, please be aware that you must first click on the entry of interest, and ensure it is highlighted in blue, before right-clicking on the entry.

21.3.1   Visualization styles and colors

Wireframe, Stick, Ball and stick, Space-filling/CPK

(◥) (◥) (◥) (●)

Four different ways of visualizing molecules by showing all atoms are provided: Wireframe, Stick, Ball and stick, Space-filling/CPK.

The visualizations are mutually exclusive meaning that only one style can be applied at a time for each selected group of atoms.

Four color schemes are available and can be accessed via right-clicking on the visualization style

icons:

- **Color by Element** Classic CPK coloring based on atom type (e.g. oxygen red, carbon gray, hydrogen white, nitrogen blue, sulfur yellow).

- **Color by Temperature** This is based on the b-factors in the PDB file and a color scale going from blue (0) over white (50) to red (100). The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **Color Different Objects** Each molecule is assigned its own random color.

- **Custom Color** The user selects molecule colors from a palette.

Backbone

()

For the molecules in the Proteins and Nucleic Acids categories, the backbone structure can be visualized in a schematic rendering, highlighting the secondary structure elements for proteins and matching base pairs for nucleic acids. The backbone visualization can be combined with any of the atom-level visualizations.

Five color schemes are available for backbone structures:

- **Color by Type** For proteins, beta sheets are blue, helices red and loops/coil gray. For nucleic acids backbone ribbons are white while the individual nucleotides are indicated in green (T/U), red (A), yellow (G), and blue (C).

- **Color by Residue Position** Rainbow color scale going from blue over green to yellow and red, following the residue number

- **Color Different Chains** Each chain/molecule is assigned its own random color.

- **Color by Backbone Temperature** based on the b-factors for the Cα atoms (the central carbon atom in each amino acid) and a color scale going from blue (0) over white (50) to red (100). The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **Custom Color** The user selects molecule colors from a palette.

Surfaces

()

Molecular surfaces can be visualized.

Five color schemes are available for surfaces:

- **Color by Charge** Charged amino acids close to the surface will show as red (negative) or blue (positive) areas on the surface, with a color gradient that depends on the distance of the charged atom to the surface.

- **Color Different Surfaces** Each surface is assigned its own random color.

- **Color by Element** Smoothed out coloring based on the classic CPK coloring of the heteroatoms close to the surface.

- **Color by Temperature** Smoothed out coloring based on the b-factors for the atoms close to the surface. The b-factors are a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder. The color scale goes from blue (0) over white (50) to red (100).

- **Custom Color** The user selects molecule colors from a palette.

A surface spanning multiple molecules can be visualized by making a custom atom group that includes all atoms from the molecules (see section 21.3.1)

Labels

(**L**)

Labels can be added to the molecules in the view by selecting an entry in the Project Tree and clicking the label button at the bottom of the Project Tree view. The color of the labels can be adjusted from the context menu by right clicking on the selected entry (which must be highlighted in blue first) or on the label button in the bottom of the Project Tree view (see figure 21.8).

Figure 21.8: *The color of the labels can be adjusted in two different ways. Either directly using the label button by right clicking the button, or by right clicking on the molecule or category of interest in the Project Tree.*

- For proteins and nucleic acids, each residue is labelled with the PDB name and number.

- For ligands, each atom is labelled with the atom name as given in the input.

- For cofactors and water, one label is added with the name of the molecule.

Labels can be removed again by clicking on the label button.

Zoom to fit

(⟵⋯⟶)

The "Zoom to fit" button can be used to automatically move a region of interest into the center of the screen. This can be done by selecting a molecule or category of interest in the **Project Tree** view followed by a click on the "Zoom to fit" button (⟵⋯⟶) at the bottom of the **Project Tree**

view (figure 21.9). Double-clicking an entry in the **Project Tree** will have the same effect.

Figure 21.9: *The "Fit to screen" button can be used to bring a particular molecule or category of molecules in focus.*

Custom atom groups

In some situations it may be relevant to use a unique visualization style or color to highlight a particular set of atoms, or to visualize only a subset of atoms from a molecule. This can be done by making an atom group selection. Atoms can be selected in different ways, as listed below. When an atom group has been created, it appears as an entry in the **Project Tree** in the category "Selections". The atoms can then be hidden or shown, and the visualization changed, just as for the molecules in the **Project Tree**.

Atom groups can be deleted from the context menu, when selecting them in the Project Tree.

How to select a particular group of atoms

A group of atoms can be selected in different ways. Brown spheres will indicate which atoms are selected in the 3D view. The selection will appear as the entry "Current" in the Selections category in the **Project Tree**. To convert the current selection into an atom group, select the Current entry in the **Project Tree**, and from the right-click context menu choose 'Create Group from Selection' to make an atom group with exactly the selected atoms, or choose 'Create Group from Selection plus Context' to include whole residues or molecules in the atom group.

- **Double click to select** Click on an atom to select it. When you double click on an atom that belongs to a residue in a protein or in a nucleic acid chain, the entire residue will be selected. For small molecules, the entire molecule will be selected.

- **Adding atoms to a selection** Holding down Ctrl while picking atoms, will pile up the atoms in the selection. All atoms in a molecule or category from the Project Tree, can be added

Figure 21.10: *An atom group that has been highlighted by adding a unique visualization style.*

to the "Current" selection by choosing "Add to Selection" in the context menu. Similarly, whole molecules can be removed from the current selection via the context menu.

- **Spherical selection** Hold down the shift-key, click on an atom and drag the mouse away from the atom. Then a sphere centered on the atom will appear, and all visible atoms inside the sphere will be selected. The status bar (lower right corner) will show the radius of the sphere.

- **Show Sequence** Another option is to select protein or nucleic acid entries in the Project Tree, and click the 'Show Sequence' button found below the Project Tree. A split-view will appear with a sequence list editor for each of the sequence data types (Protein, DNA, RNA) (figure 21.11). If you go to the protein (or nucleic acid) sequence that was opened with the "Show Sequence" button and select a region in the sequence, the selected residues will show up as the "Current" selection in the 3D view and the Project Tree view. Notice that the link between the 3D view and the sequence editor is lost if either window is closed, or if the sequences are modified.

21.3.2   Project settings

A number of general settings can be adjusted from the **Side Panel**. Personal settings can be saved by clicking in the lower right corner of the **Side Panel** (⬚). This is described in detail in section 11.6.

Project Tree Actions

Just below the Project Tree, the following action is available

- **Show Sequence** Select molecules which have sequences associated (Protein, DNA, RNA) in the Project Tree, and click this button. Then, a split-view will appear with a sequence list editor for each of the sequence data types (Protein, DNA, RNA). This is described in section 21.5.
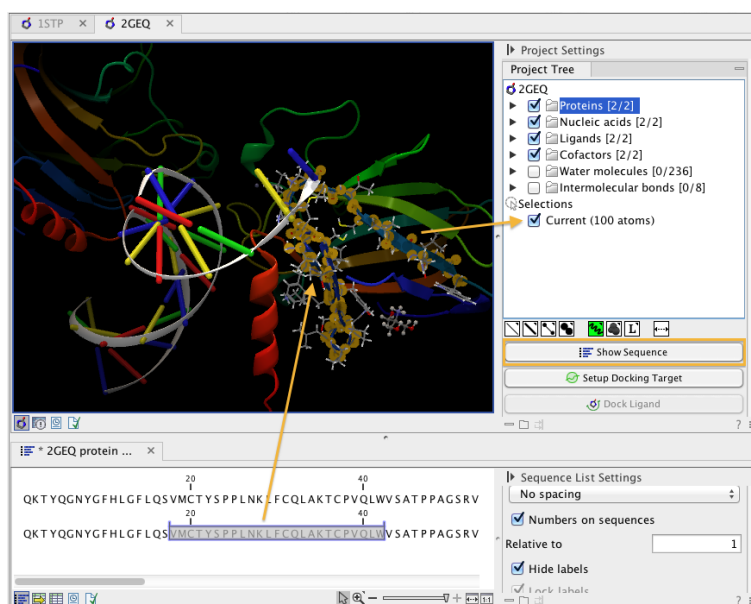
Figure 21.11: *The protein sequence in the split view is linked with the protein structure. This means that when a part of the protein sequence is selected, the same region in the protein structure will be selected.*

Property viewer

The Property viewer, found in the Side Panel, lists detailed information about the atoms that the mouse hovers over. For all atoms the following information is listed:

- **Name** The particular atom name, if given in input.

- **Element** The element type (C, N, O,...).

- **Hybridization** The atom hybridization assigned to the atom.

- **Charge** The atomic charge as given in the input file. If charges are not given in the input file, some charged chemical groups are automatically recognized and a charge assigned.

- **Molecule** The name of the molecule the atom is part of.

- **Chain** For proteins and nucleic acids, the name of the chain the atom belongs to is listed.

- **Residue** For proteins and nucleic acids, the name and number of the residue the atom belongs to is listed.

For atoms in molecules imported from a PDB file, extra information is given:

- **Temperature** Here is listed the b-factor assigned to the atom in the PDB file. The b-factor is a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.

- **PDB index** Each atom listed in the PDB file is given an index number, which is listed in the second column of the ATOM or HETATOM entries in the PDB file.

- **Source line** The line number in the PDB text file where the atom information appears.

If an atom is selected, the Property view will be frozen with the details of the selected atom shown. If then a second atom is selected (by holding down Ctrl while clicking), the distance between the two selected atoms is shown. If a third atom is selected, the angle for the second atom selected is shown. If a fourth atom is selected, the dihedral angle measured as the angle between the planes formed by the three first and three last selected atoms is given.

Figure 21.12: *Selecting two, three, or four atoms will display the distance, angle, or dihedral angle, respectively.*

If a molecule is selected in the Project Tree, the Property view shows information about this molecule:

- **Atoms** Number of atoms in the molecule.

- **Weight** The weight of the molecule in Daltons.

Visualization settings

Under "Visualization" four options exist:

- **Hydrogens** Hydrogen atoms can be shown (Show all hydrogens), hidden (Hide all hydrogens) or partially shown (Show only polar hydrogens).

- **Fog** "Fog" is added to give a sense of depth in the view. The strength of the fog can be adjusted or it can be disabled.

- **3D projection** The view is opened up towards the viewer, with a "Perspective" 3D projection. The field of view of the perspective can be adjusted, or the perspective can be disabled by selecting an orthographic 3D projection.

- **Coloring** The background color can be selected from a color palette by clicking on the colored box.

21.4   Snapshots of the molecule visualization

To save the current view as a picture, right-click in the **View Area** and select "File" and "Export Graphics". Another way to save an image is by pressing the "Graphics" button in the Workbench toolbar (![icon]). Next, select the location where you wish to save the image, select file format (PNG, JPEG, or TIFF), and provide a name, if you wish to use another name than the default name.

21.5 Sequences associated with the molecules

From the Side Panel, sequences associated with the molecules in the Molecule Project can be opened as separate objects by selecting molecule entries in the Project Tree and clicking the button labeled "Show Sequence" (figure 21.13). This will generate a sequence list for each selected sequence type (protein, DNA, RNA). The sequence list can be used to select atoms in the Molecular Project as described in (section 21.3.1). The sequence list can also be saved as an independent object and used as input for sequence analysis tools.
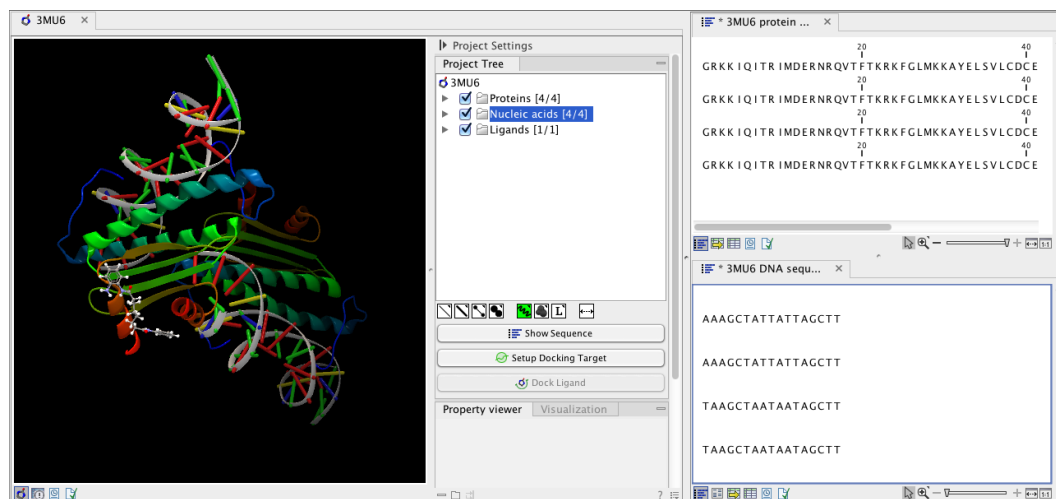
Figure 21.13: *All protein chain sequences as well as DNA sequences of interacting DNA are shown as individual sequences.*

21.6 Troubleshooting 3D graphics errors

The 3D viewer uses OpenGL graphics hardware acceleration in order to provide the best possible experience. If you experience any graphics problems with the 3D view, please make sure that the drivers for your graphics card are up-to-date.

If the problems persist after upgrading the graphics card drivers, it is possible to change to a rendering mode, which is compatible with a wider range of graphic cards. To change the graphics mode go to Edit in the menu bar, select "Preferences", Click on "View", scroll down to the bottom and find "Molecule Project 3D Editor" and uncheck the box "Use modern OpenGL rendering".

Finally, it should be noted that certain types of visualization are more demanding than others. In particular, using multiple molecular surfaces may result in slower drawing, and even result in the graphics card running out of available memory. Consider creating a single combined surface (by using a selection) instead of creating surfaces for each single object. For molecules with a large number of atoms, changing to wireframe rendering and hiding hydrogen atoms can also greatly improve drawing speed.

21.7 Updating old structure files

As the 3D Molecule Viewer has been completely redesigned, it is necessary to update old structure files. To update existing structure files, double click on the name in the **Navigation Area**. This will bring up the dialog shown in figure 21.14, which via the "Download from PDB..." button gives access to downloading the specific structure in PDB format.
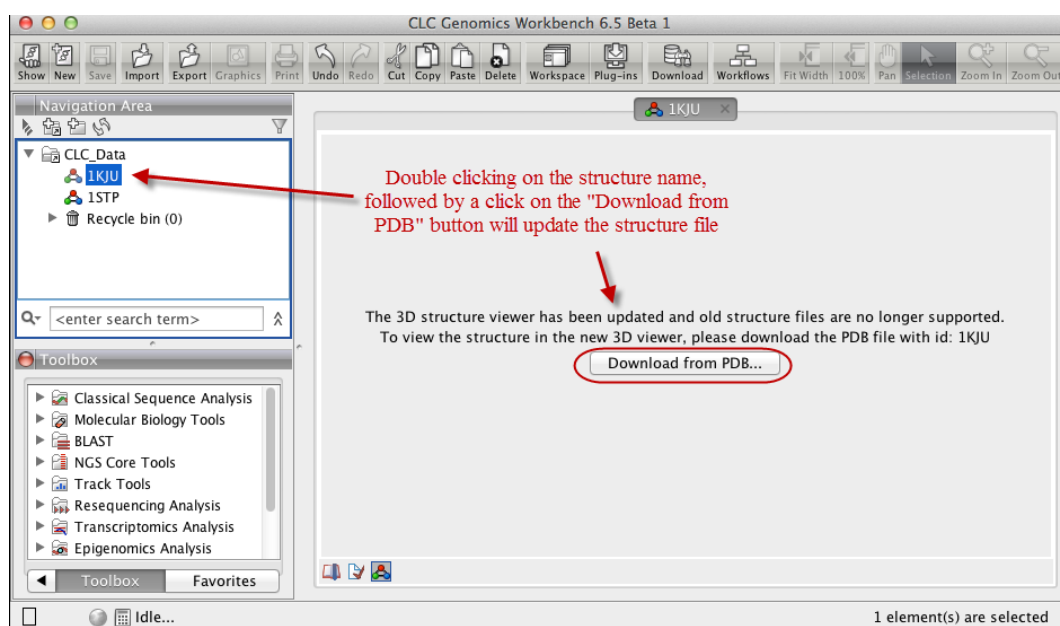
Figure 21.14: *Old structure files are not supported by the new 3D Molecule Viewer and must be updated.*

Chapter 22

General sequence analyses

Contents

CLC Main Workbench offers different kinds of sequence analyses, which apply to both protein and DNA. The analyses are described in this chapter.

22.1 Extract sequences

This tool allows the extraction of sequences from other types of data in the Workbench, such as sequence lists or alignments. The data types you can extract sequences from are:

- Alignments ()

- BLAST result (▤)

- BLAST overview tables (▤)

- sequence lists (▤)

- Contigs and read mappings (▤)

- Read mapping tables (▤)

- Read mapping tracks (▤)

- RNA-Seq mapping results (▤)

Note! When the Extract Sequences tool is run via the Workbench toolbox, **all** sequences are extracted from the data used as input. If only a subset of the sequences is desired, for example, the reads from just a small area of a mapping, or the sequences for only a few blast results, then a data set containing just this subsection or subset should be created and the Extract Sequences tool should be run on that.

For extracting a subset of a mapping, please see section 26.8.6 that describes the function "Extract from Selection" that also can be selected from the right click menu (see figure 22.1).

For extracting a subset of a sequence list, you can highlight the sequences of interest in the table view of the sequence list, right click on the selection and launch the Extract Sequences tool.

The Extract Sequences tool can be launched via the Toolbox menu:

Toolbox | General Sequence Analysis (▤)| Extract Sequences (▤)

Alternatively, on all the data types listed above except sequence lists, the option to run this tool appears by right clicking in the relevant area: a row in a table or in the read area of mapping data. An example is shown in figure 22.1.

Please note that for mappings, only the read sequences are extracted. Reference and consensus sequences are not extracted using this tool. Similarly, when extracting sequences from BLAST results, the sequence hits are extracted, not the original query sequence or a consensus sequence.
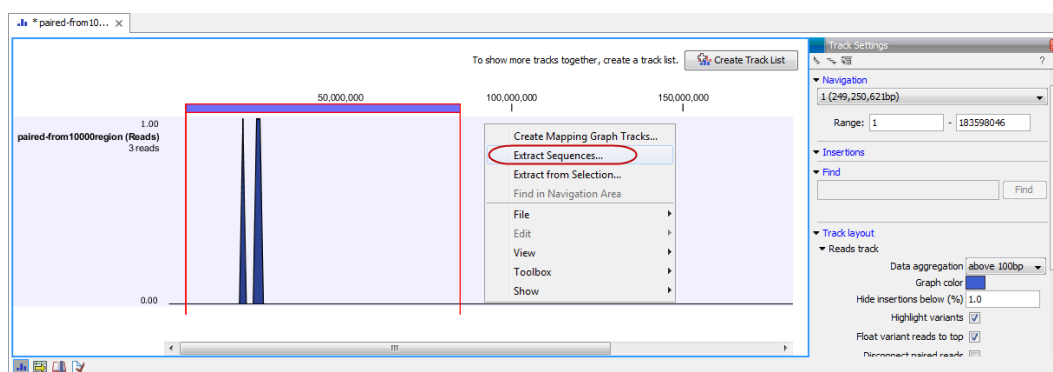
Figure 22.1: *Right click somewhere in the reads track area and select "Extract Sequences".*

The dialog allows you to select the **Destination**. Here you can choose whether the extracted sequences should be extracted as single sequences or placed in a new sequence list. For most
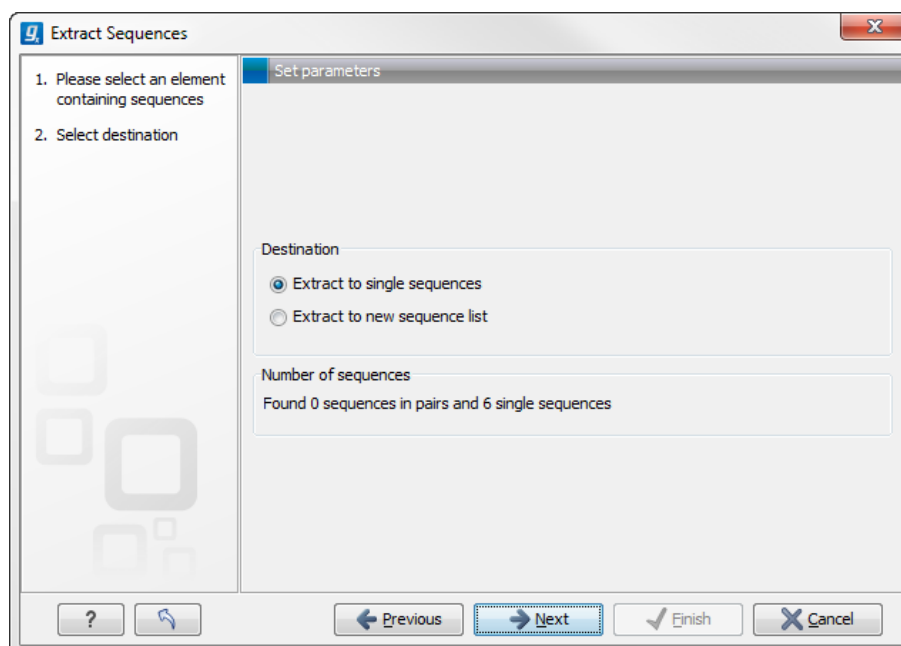
Figure 22.2: *Choosing whether the extracted sequences should be placed in a new list or as single sequences.*

data types, it will make most sense to choose to extract the sequences into a sequence list. The exception to this is when working with a sequence list, where choosing to extract to a sequence list would create a copy of the same sequence list. In this case, the other option would generally be chosen. This would then result in the generation of individual sequence objects for each sequence in the sequence list.

Below these options, in the dialog, you can see the number of sequences that will be extracted.

22.2   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 Analysis ( )| Shuffle Sequence ( )**

or   **right-click a sequence | Toolbox | General Sequence Analysis ( )| Shuffle Sequence ( )**

This opens the dialog displayed in figure 22.3:

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 22.4:   For nucleotides, the following parameters can be set:
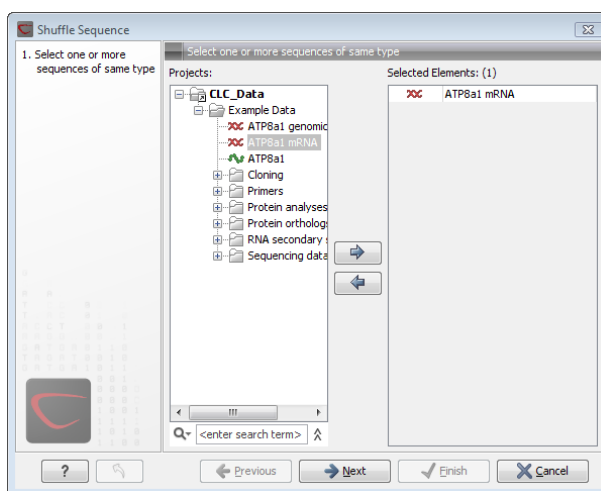
Figure 22.3: *Choosing sequence for shuffling.*
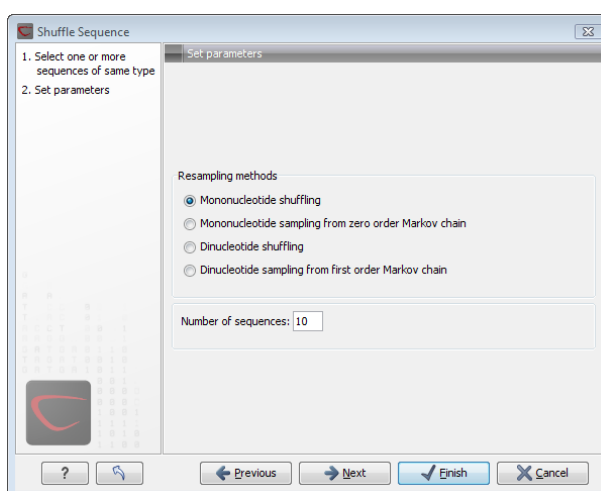
Figure 22.4: *Parameters for shuffling.*

- **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 15.2). 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.

22.3   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.

22.3.1   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 Analysis ( )| Create Dot Plot ( )**

>   or   **Select one or two sequences in the Navigation Area | Toolbox in the Menu Bar | General Sequence Analysis ( )| Create Dot Plot ( )**

>   or   **Select one or two sequences in the Navigation Area | right-click in the Navigation Area | Toolbox | General Sequence Analysis ( )| Create Dot Plot ( )**

This opens the dialog shown in figure 22.5.

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 22.6.

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
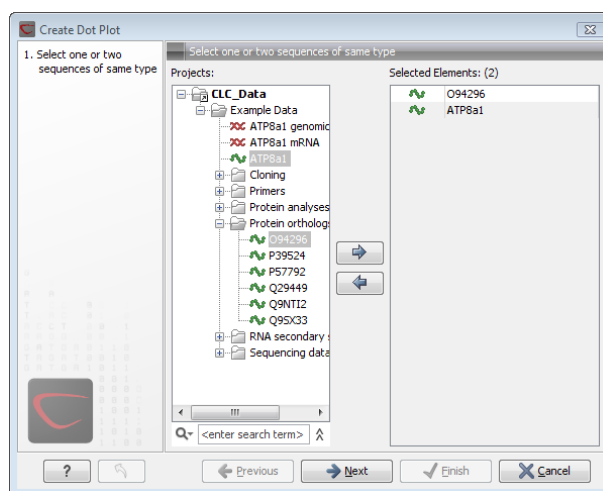
Figure 22.5: *Selecting sequences for the dot plot.*

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 15.2). If not, click **Finish**.

22.3.2   View dot plots

A view of a dot plot can be seen in figure 22.7. You can select **Zoom in** (⊕) in the Toolbar and click the dot plot to zoom in to see the details of particular areas.

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
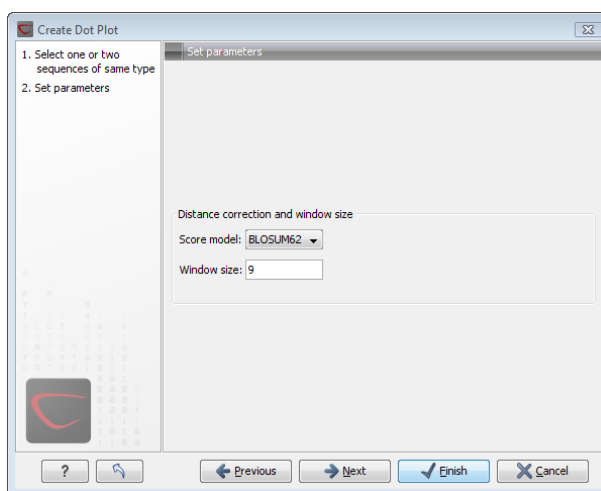
Figure 22.6: *Setting the dot plot parameters.*
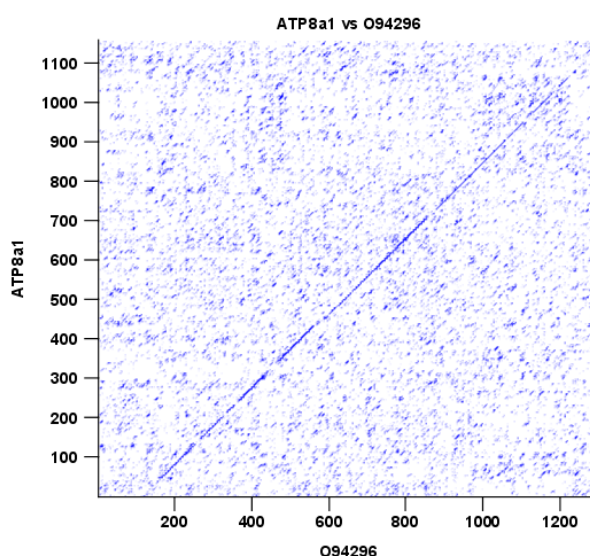
Figure 22.7: *A view is opened showing the dot plot.*

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 22.7.

22.3.3   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
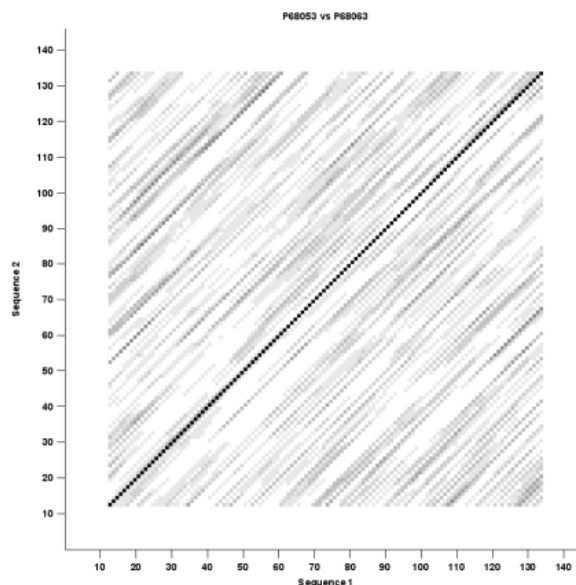
Figure 22.8: *Dot plot with inverted colors, practical for printing.*

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 22.9 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.
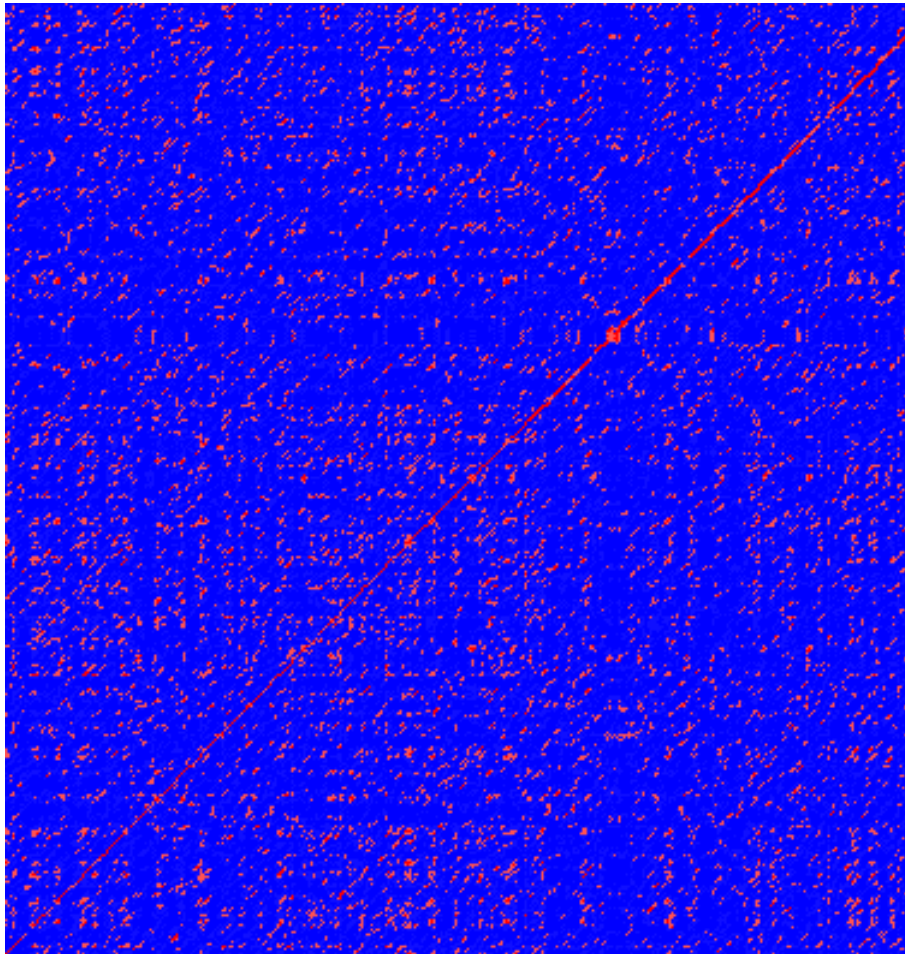
Figure 22.9: *Dot plot of DQ232610 vs. DQ023146 (Influenza A virus nucleoproteins) showing and overall similarity*

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.
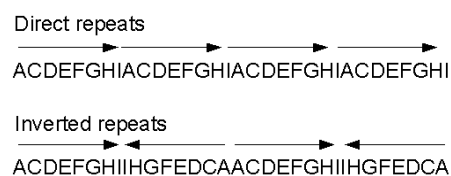
Figure 22.10: *Direct and inverted repeats shown on an amino acid sequence generated for demonstration purposes.*

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 22.11 you can see a sequence with repeats.

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 22.12. In this figure, three frame
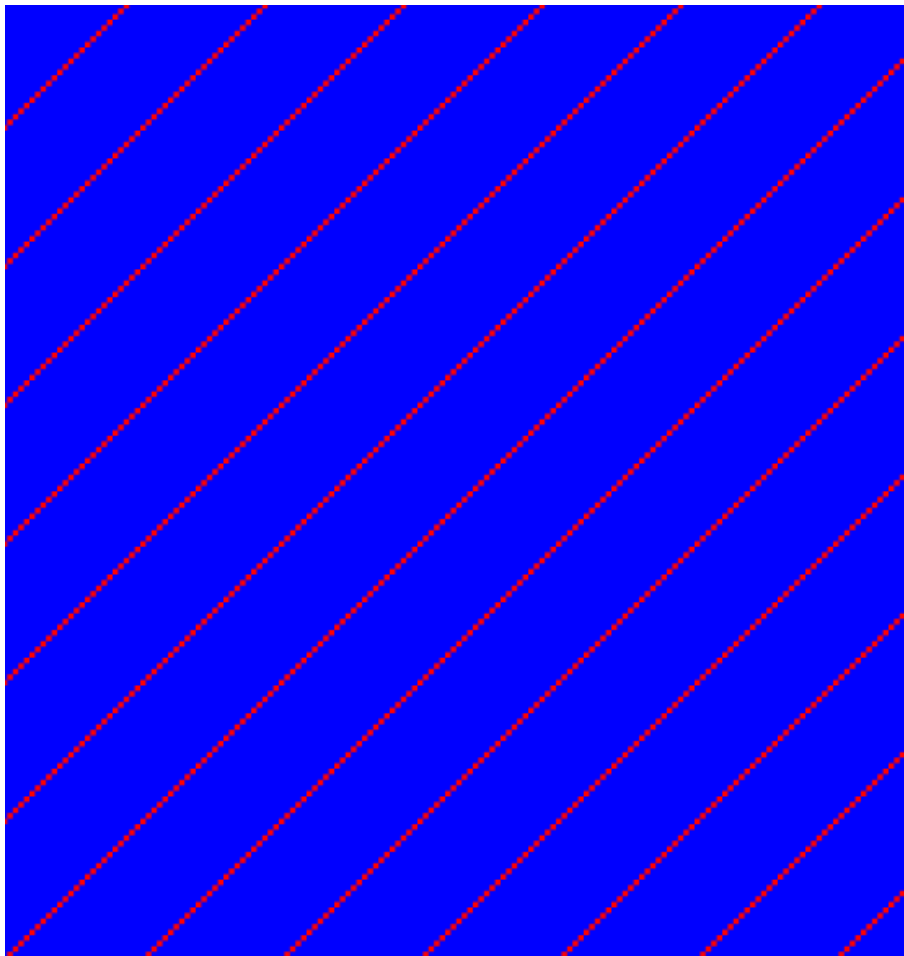
Figure 22.11: *The dot plot of a sequence showing repeated elements. See also figure 22.10.*

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

In dot plots you can see an inversion of sequence as contrary diagonal to the diagonal showing similarity. In figure 22.13 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 22.14 is a square shows the low-complexity region of this sequence.
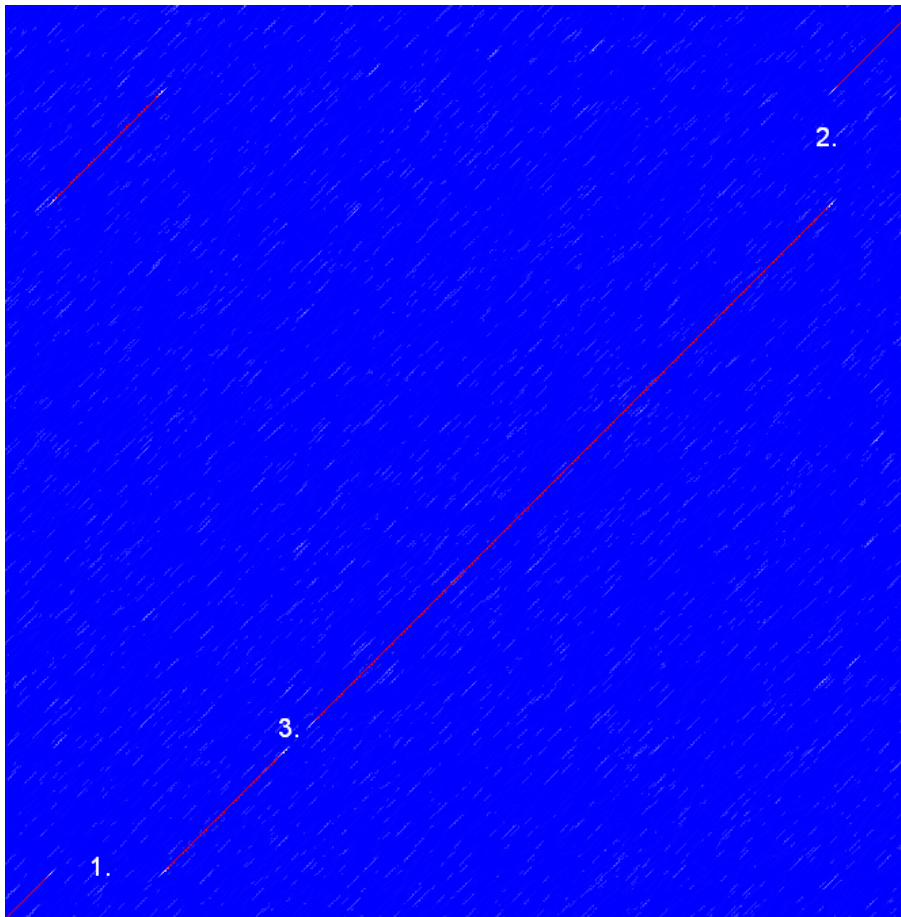
Figure 22.12: *This dot plot show various frame shifts in the sequence. See text for details.*

Creative Commons License

22.3.4   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
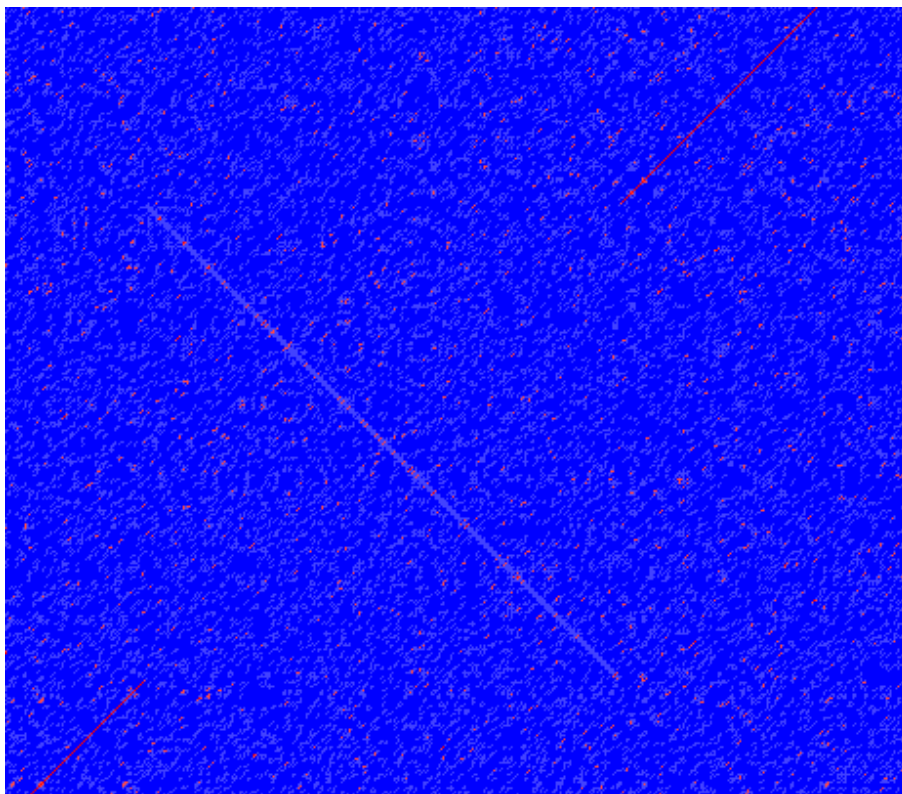
Figure 22.13: *The dot plot showing an inversion in a sequence. See also figure 22.10.*

acids can be modeled by a scoring matrix also refereed to as a substitution matrix. See an example of a scoring matrix in table 22.1. 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].

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 22.15).

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
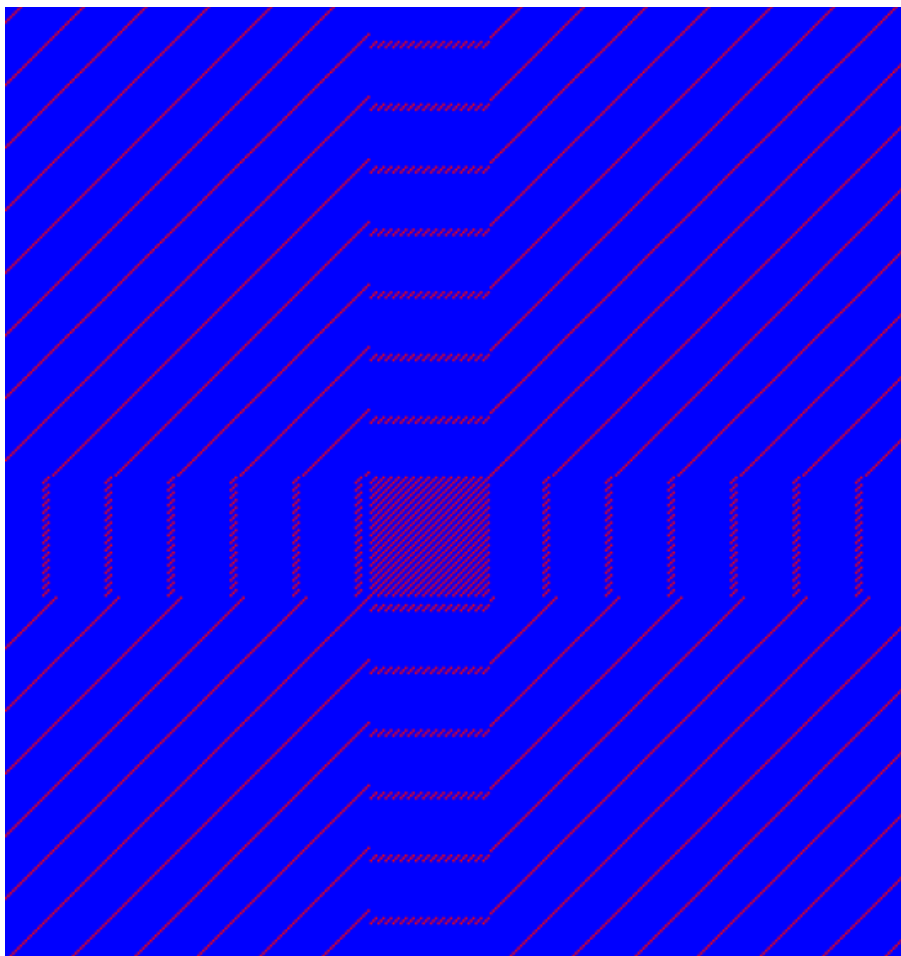
Figure 22.14: *The dot plot showing a low-complexity region in the sequence. The sequence is artificial and low complexity regions do not always show as a square.*

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 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.

	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 22.1: **The BLOSUM62 matrix**. A tabular view of the BLOSUM62 matrix containing all possible substitution scores [Henikoff and Henikoff, 1992].

- 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 22.15) 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.

Figure 22.15: *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.*

Other useful resources

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

All CLC bio's scientific articles are licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. You are free to copy, distribute, display, and use the work for educational purposes, under the following conditions: You must attribute the work in its original form and "CLC bio" has to be clearly labeled as author and provider of the work. You may not use this work for commercial purposes. You may not alter, transform, nor build upon this work.

See http://creativecommons.org/licenses/by-nc-nd/2.5/ for more information on how to use the contents.

22.4   Local complexity plot

In *CLC Main 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 Analysis ( )** | **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 22.16 shows an example of a local complexity plot.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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.

22.5   Sequence statistics

CLC Main 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 Analysis ( )** | **Create Sequence Statistics ( )**
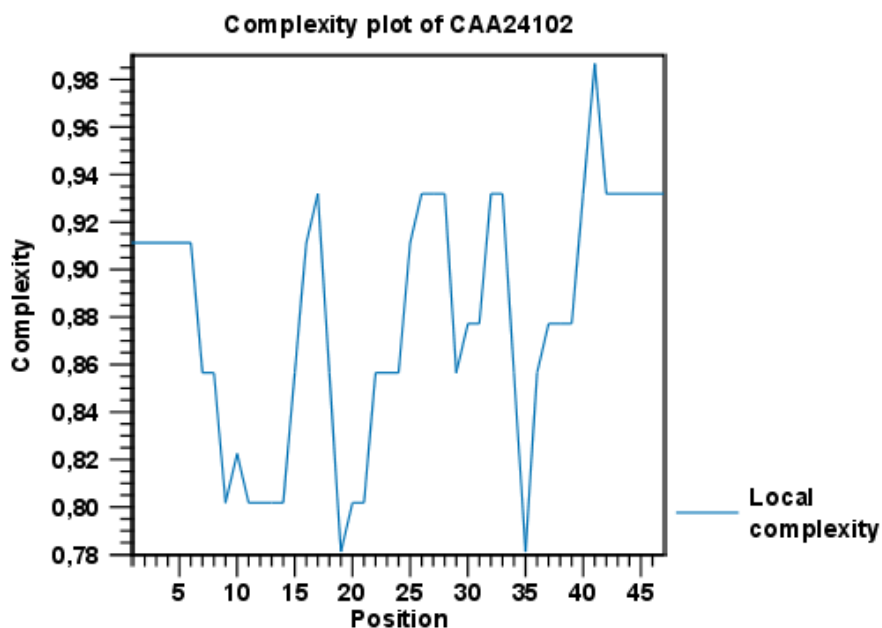
Figure 22.16: *An example of a local complexity plot.*

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 22.17.
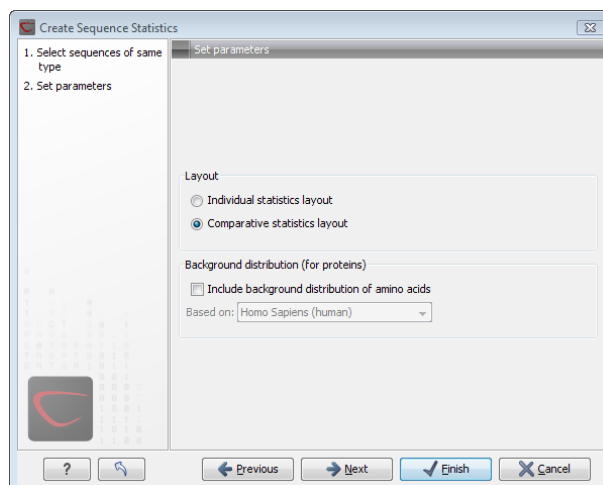
Figure 22.17: *Setting parameters for the sequence statistics.*

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 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 15.2). If not, click **Finish**. An example of protein sequence statistics is shown in figure 22.18.

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 22.18: *Example of protein 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. This is calculated like this: $sum_{unitsinsequence}(weight(unit)) - links * weight(H2O)$ where `links` is the sequence length minus one and `units` are amino acids. The atomic composition is defined the same way.
    - 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. This is calculated like this: $sum_{units in sequence}(weight(unit)) - links * weight(H2O)$ where `links` is the sequence length minus one for linear sequences and sequence length for circular molecules. The `units` are monophosphates. Both the weight for single- and double stranded molecules are includes. The atomic composition is defined the same way.

- Atomic composition

- Nucleotide distribution table

- Nucleotide distribution histogram

- Annotation table

- Counts of di-nucleotides

- Frequency of di-nucleotides

If nucleotide sequences are used as input, and these are annotated with CDS, a section on Codon statistics for Coding Regions is included.

A short description of the different areas of the statistical output is given in section 22.5.1.

22.5.1   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.

$$Aliphatic index = X(Ala) + a * X(Val) + b * X(Leu) + b * (X)Ile$$

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].

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 22.2). If leucine is found N-terminally in mammalian proteins the estimated half-life is 5.5 hours.

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 22.2: **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*.

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

All CLC bio's scientific articles are licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. You are free to copy, distribute, display, and use the work for educational purposes, under the following conditions: You must attribute the work in its original form and "CLC bio" has to be clearly labeled as author and provider of the work. You may not use this work for commercial purposes. You may not alter, transform, nor build upon this work.

See http://creativecommons.org/licenses/by-nc-nd/2.5/ for more information on how to use the contents.

22.6   Join sequences

CLC Main 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 22.19.
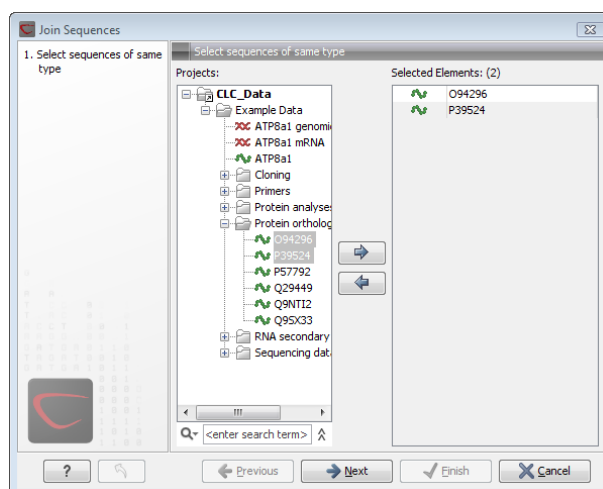
Figure 22.19: *Selecting two sequences to be joined.*

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 22.20.

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.
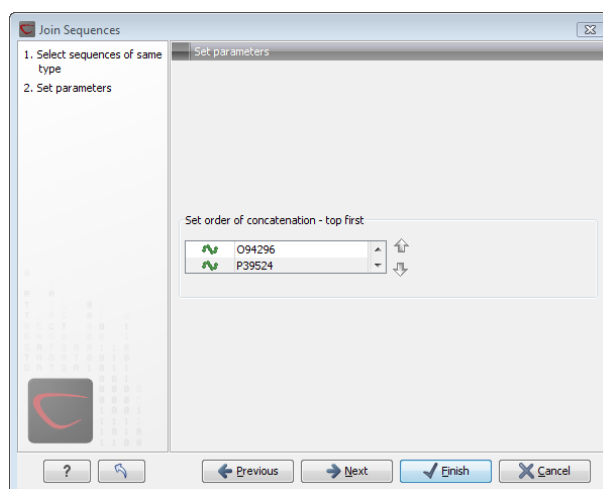
Figure 22.20: *Setting the order in which sequences are joined.*

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

The result is shown in figure 22.21.
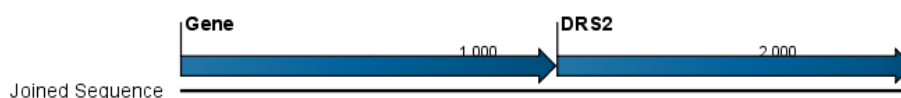
Figure 22.21: *The result of joining sequences is a new sequence containing the annotations of the joined sequences (they each had a HBB annotation).*

22.7   Pattern Discovery

With *CLC Main 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 Analysis ( )| Pattern Discovery ( )**

or **right-click DNA or protein sequence(s) | Toolbox | General Sequence Analysis ( )| 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 22.22).

In order to search unknown sequences with an already existing model:
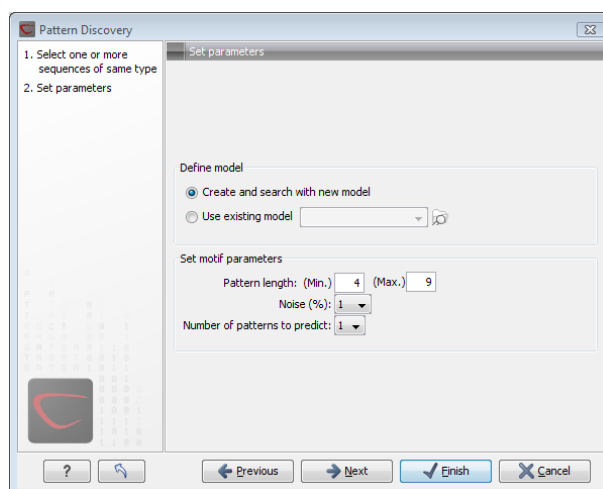
Figure 22.22: *Setting parameters for the pattern discovery. See text for details.*

Select to use an already existing model which is seen in figure 22.22. Models are represented with the following icon in the **Navigation Area** (▨).

22.7.1   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 22.22.

- **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 15.2). If not, click **Finish**. This will open a view showing the patterns found as annotations on the original sequence (see figure 22.23). If you have selected several sequences, a corresponding number of views will be opened.

Figure 22.23: *Sequence view displaying two discovered patterns.*

22.7.2 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.

22.8 Motif Search

CLC Main 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 28.3.1) . 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.

22.8.1 Dynamic motifs

In the **Side Panel** of sequence views, there is a group called **Motifs** (see figure 22.24).

The Workbench will look for the listed motifs in the sequence that is open and by clicking the

Figure 22.24: *Dynamic motifs in the Side Panel.*

Figure 22.25: *Showing dynamic motifs on the sequence.*

check box next to the motif it will be shown in the view as illustrated in figure 22.25.

This case shows the CMV promoter primer sequence which is one of the pre-defined motifs in *CLC Main 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 22.26.

Figure 22.26: *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 matches in N-regions for simple motifs**. 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 and if a one residue in a motif matches to an N, it will be treated as a mismatch.

The list of motifs shown in figure 22.24 is a pre-defined list that is included with the *CLC Main Workbench*. You can define your own set of motifs to use instead. In order to do this, you can either click on the **Add Motif** button in the side panel (see figure 22.24) and directly define and add motifs of choice as illustrated in figure 22.30. Alternatively, you can create and save a **Motif list** () (see section 22.8.4). Subsequently, in the sequence view click the **Manage Motifs** button in the side panel which will bring up the dialog shown in figure 22.27.
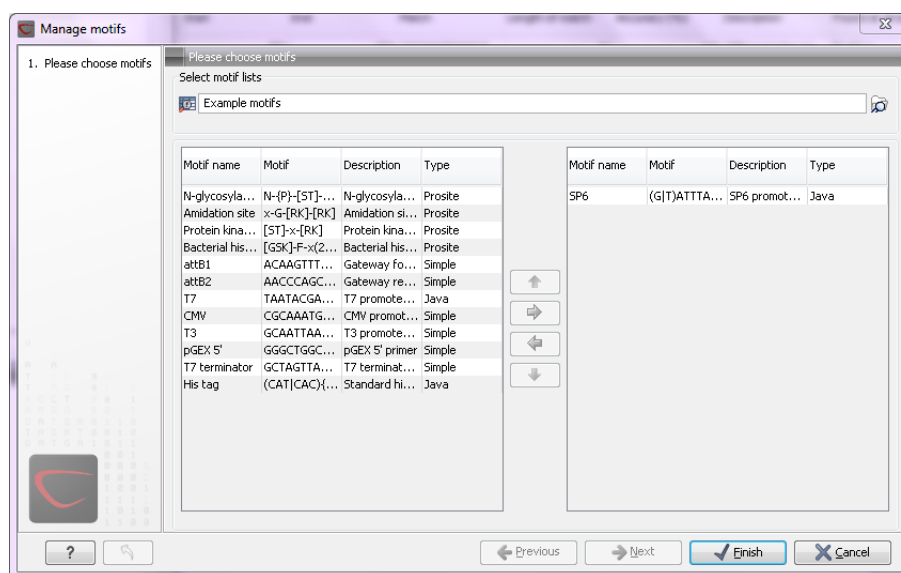
Figure 22.27: *Managing the motifs to be shown.*

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**.

22.8.2   Motif search from the Toolbox

The dynamic motifs described in section 22.8.1 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 Analysis** () | **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 22.28).
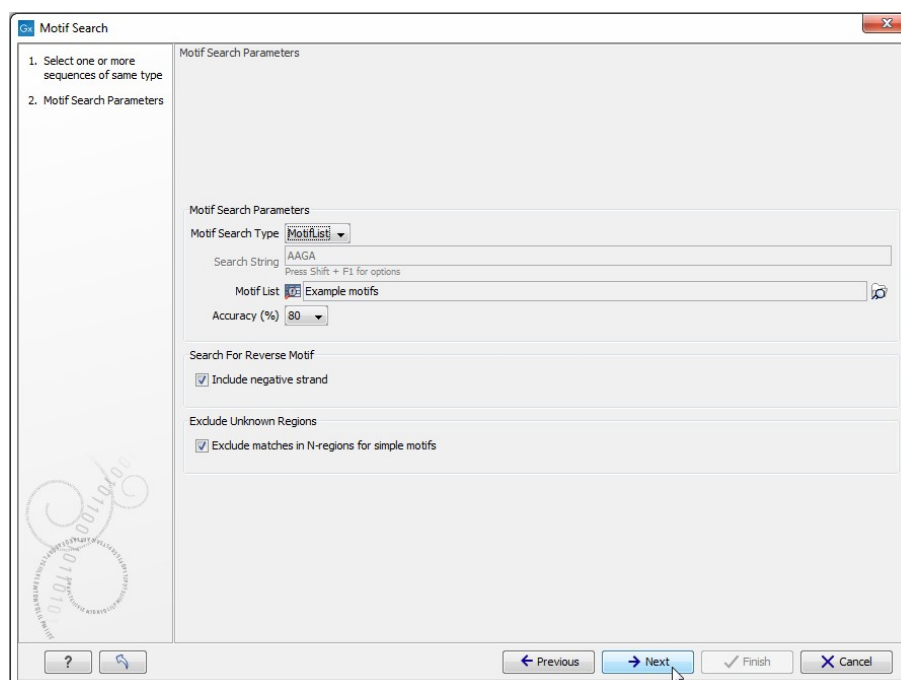
Figure 22.28: *Setting parameters for the motif search.*

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 22.8.3.

  - 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 22.8.4).

- **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 22.8.3. 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** to adjust how to handle the results and then 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 22.29).

- **Create table**. This will create an overview table of all the motifs found for all the input sequences.
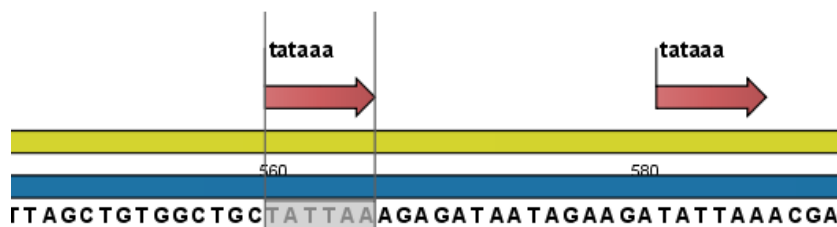
Figure 22.29: *Sequence view displaying the pattern found. The search string was 'tataaa'.*

22.8.3 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*.

[ˆA-M] will match any character except those between *A* and *M* (Excluding). You can also put single characters between the brackets: The expression *[ˆAG]* matches any character except *A* and *G*.

[A-Z&&[ˆM-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&&[ˆCG]]* 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 ˆ restricts the search to the beginning of your sequence. For example, if you search through a sequence with the regular expression ˆ*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][ˆAC]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.[ˆA]$* 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*.

22.8.4   Create motif list

CLC Main 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 Analysis ( )| 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 22.30.

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.
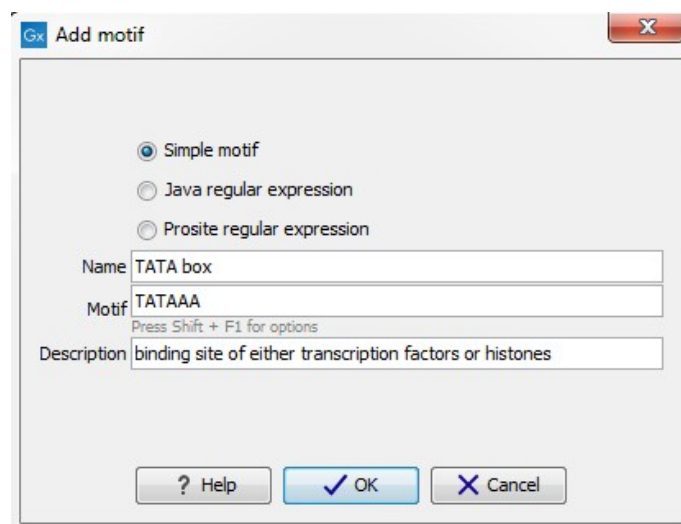
Figure 22.30: *Entering a new motif in the list.*

- **Motif**. The actual motif. See section 22.8.2 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 22.8.2.

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 22.8).

Chapter 23

Nucleotide analyses

Contents

CLC Main Workbench offers different kinds of sequence analyses, which only apply to DNA and RNA.

23.1 Convert DNA to RNA

CLC Main 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 Analysis ( )| Convert DNA to RNA ( )**

> or **right-click a sequence in Navigation Area | Toolbox | Nucleotide Analysis ( )| Convert DNA to RNA ( )**

This opens the dialog displayed in figure 23.1:

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 15.2). 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.
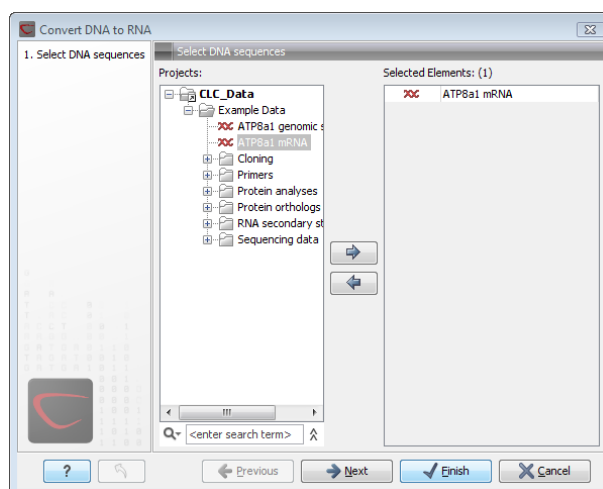
Figure 23.1: *Translating DNA to RNA.*

23.2 Convert RNA to DNA

CLC Main 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 Analysis ( )| Convert RNA to DNA ( )**

or **right-click a sequence in Navigation Area | Toolbox | Nucleotide Analysis ( )| Convert RNA to DNA ( )**

This opens the dialog displayed in figure 23.2:

Figure 23.2: *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 15.2). 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.

23.3   Reverse complements of sequences

CLC Main 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 Analysis ( )**| **Reverse Complement ( )**

> or **right-click a sequence in Navigation Area** | **Toolbox** | **Nucleotide Analysis ( )**| **Reverse Complement ( )**

This opens the dialog displayed in figure 23.3:
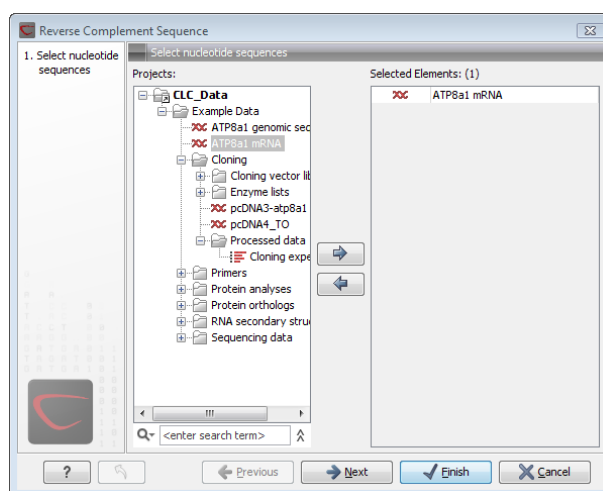
Figure 23.3: *Creating a reverse complement sequence.*

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 15.2). 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.

23.4 Reverse sequence

CLC Main Workbench is able to create the reverse of a nucleotide sequence.

Note! This is not the same as a reverse complement. If you wish to create the reverse complement, please refer to section 23.3.

> **select a sequence in the Navigation Area | Toolbox in the Menu Bar | Nucleotide Analysis (🖼️)| Reverse Sequence (🔧)**

This opens the dialog displayed in figure 23.4:
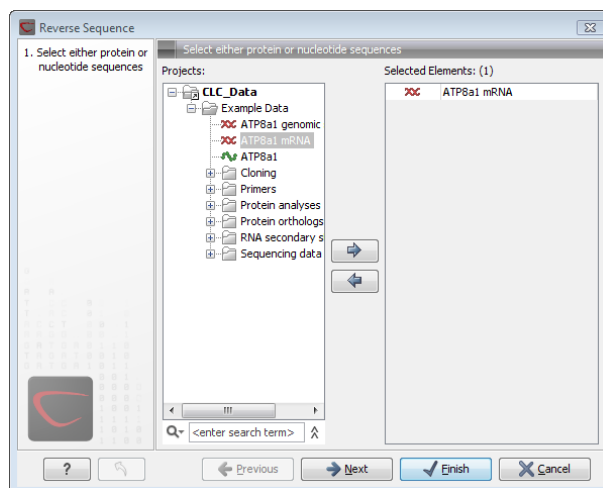
Figure 23.4: *Reversing a sequence.*

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 15.2). 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 23.3.

23.5 Translation of DNA or RNA to protein

In *CLC Main 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 Analysis (🖼️)| Translate to Protein (〰️)**

or **right-click a nucleotide sequence | Toolbox | Nucleotide Analysis (🖼️)| Translate to Protein (〰️)**

This opens the dialog displayed in figure 23.5:

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in
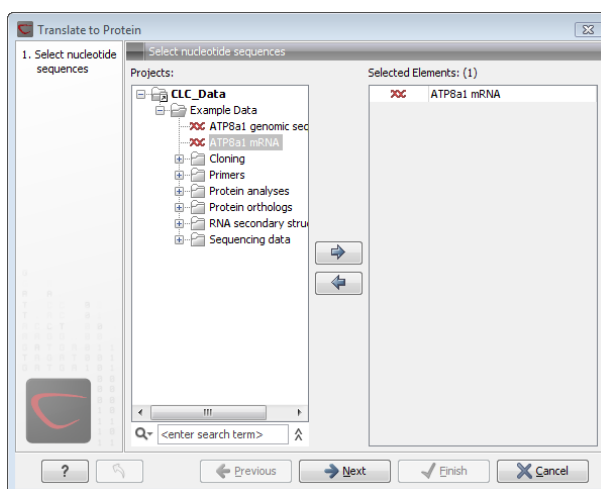
Figure 23.5: *Choosing sequences for translation.*

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 23.6:
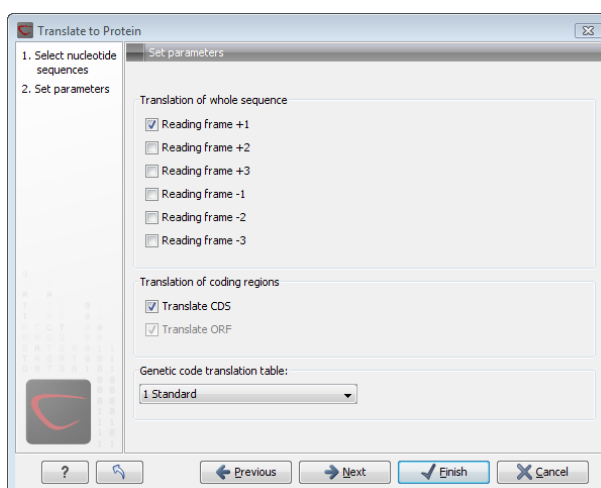
Figure 23.6: *Choosing +1 and +3 reading frames, and the standard translation table.*

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 15.2). 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.

23.5.1 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 23.6).

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.

23.6 Find open reading frames

The *CLC Main Workbench* **Find Open Reading Frames** function can be used to find all open reading frames (ORF) in a sequence, or, by choosing particular start codons to use, it can be used as a rudimentary gene finder. ORFs identified will be shown as annotations on the sequence. You have the option of choosing a translation table, the start codons to use, minimum ORF length as well as a few other parameters. These choices are explained in this section.

To find open reading frames:

> **select a nucleotide sequence | Toolbox in the Menu Bar | Nucleotide Analysis ( )| Find Open Reading Frames ( )**

> or **right-click a nucleotide sequence | Toolbox | Nucleotide Analysis ( )| Find Open Reading Frames ( )**

This opens the dialog displayed in figure 23.7:
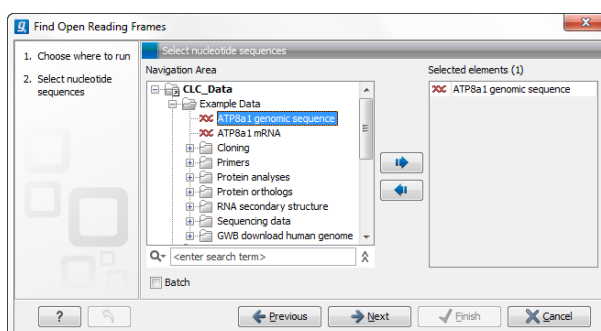
Figure 23.7: *Create Reading Frame dialog.*

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**.

23.6.1 Open reading frame parameters

This opens the dialog displayed in figure 23.8:
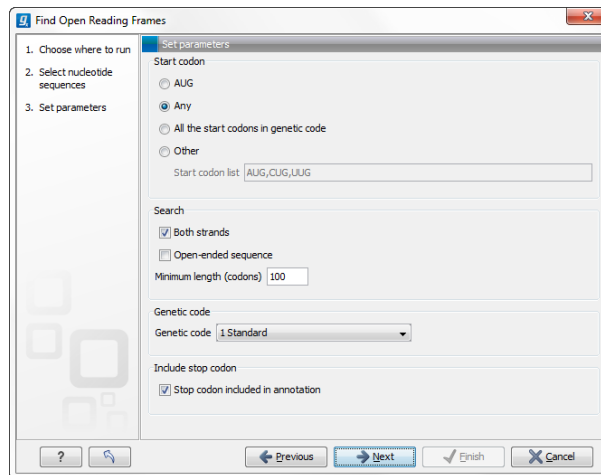
Figure 23.8: *Create Reading Frame dialog.*

The adjustable parameters for the search are:

- **Start codon**:

    - **AUG**. Most commonly used start codon.

    - **Any**. Find all open reading frames of specified length. Any combination of three bases that is not a stop-codon is interpreted as a start codon, and translated according to the specified genetic code.

    - **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 23.9).
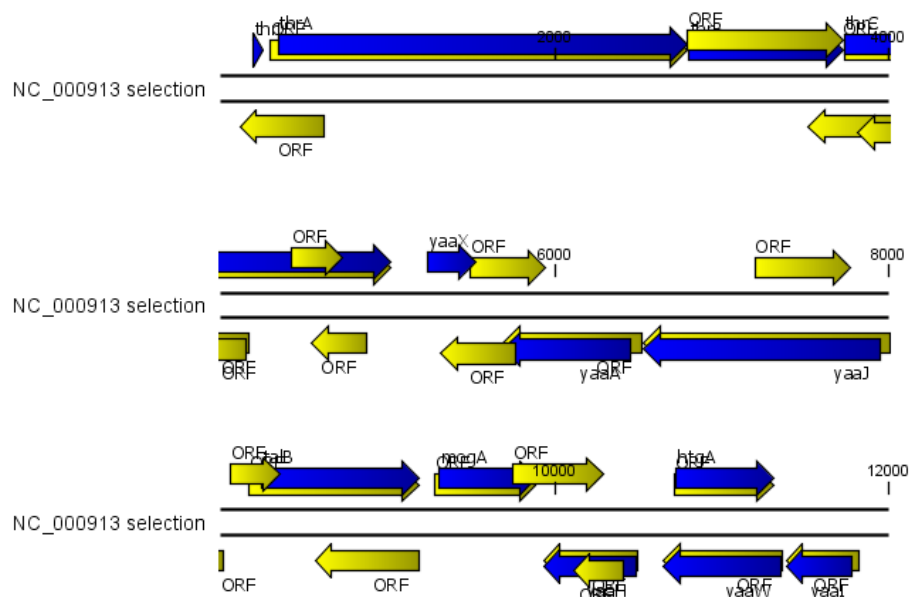
Figure 23.9: *The first 12,000 positions of the* E. coli *sequence NC_000913 downloaded from GenBank. The blue (dark) annotations are the genes while the yellow (brighter) annotations are the ORFs with a length of at least 100 amino acids. On the positive strand around position 11,000, a gene starts 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 15.2). 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 24

Protein analyses

Contents

CLC Main Workbench offers a number of analyses of proteins as described in this chapter.

24.1   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 24.1.3). 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 plugin* using the plugin manager, see section 1.7.1.

When the plugin is downloaded and installed, you can use it to predict signal peptides:

> **Select a protein sequence** | **Toolbox in the Menu Bar** | **Protein Analysis (🖼️)**| **Signal Peptide Prediction (🦋)**

> or   **right-click a protein sequence** | **Toolbox** | **Protein Analysis (🖼️)**| **Signal Peptide Prediction (🦋)**
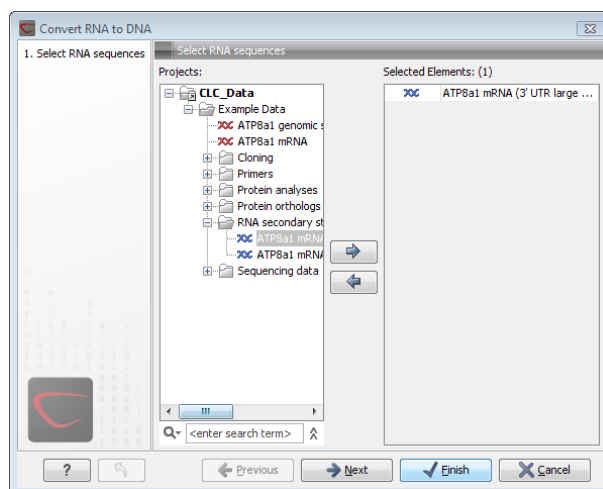
If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** to set parameters for the SignalP analysis.

24.1.1   Signal peptide prediction parameter settings

It is possible to set different options prior to running the analysis (see figure 24.1). An organism type should be selected. The default is eukaryote.

- Eukaryote (default)

- Gram-negative bacteria

- Gram-positive bacteria

In addition, you can choose between two methods for prediction: Hidden Markov Model and Neural Network.

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:
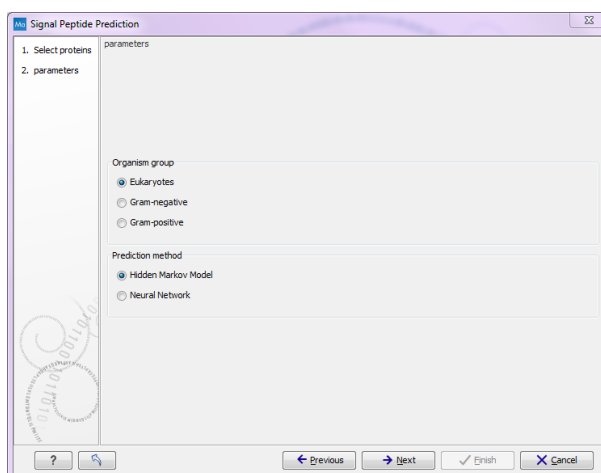
Figure 24.1: *Setting the parameters for signal peptide prediction.*

- Add annotations to sequence

- Create table

- Text

Click **Next** to adjust how to handle the results, then click **Finish**.

24.1.2 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 24.2).
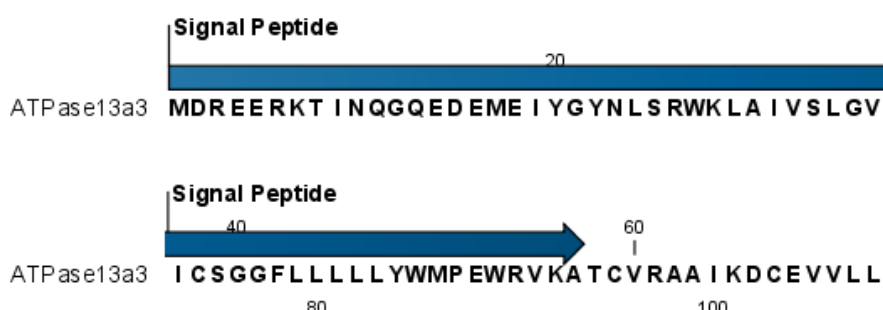
Figure 24.2: *N-terminal signal peptide shown as annotation on the sequence.*

Additional notes can be added through the **Edit annotation** ( ) right-click mouse menu. See section 17.3.3.

Undesired annotations can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 17.3.5.

24.1.3 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 24.3.

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/
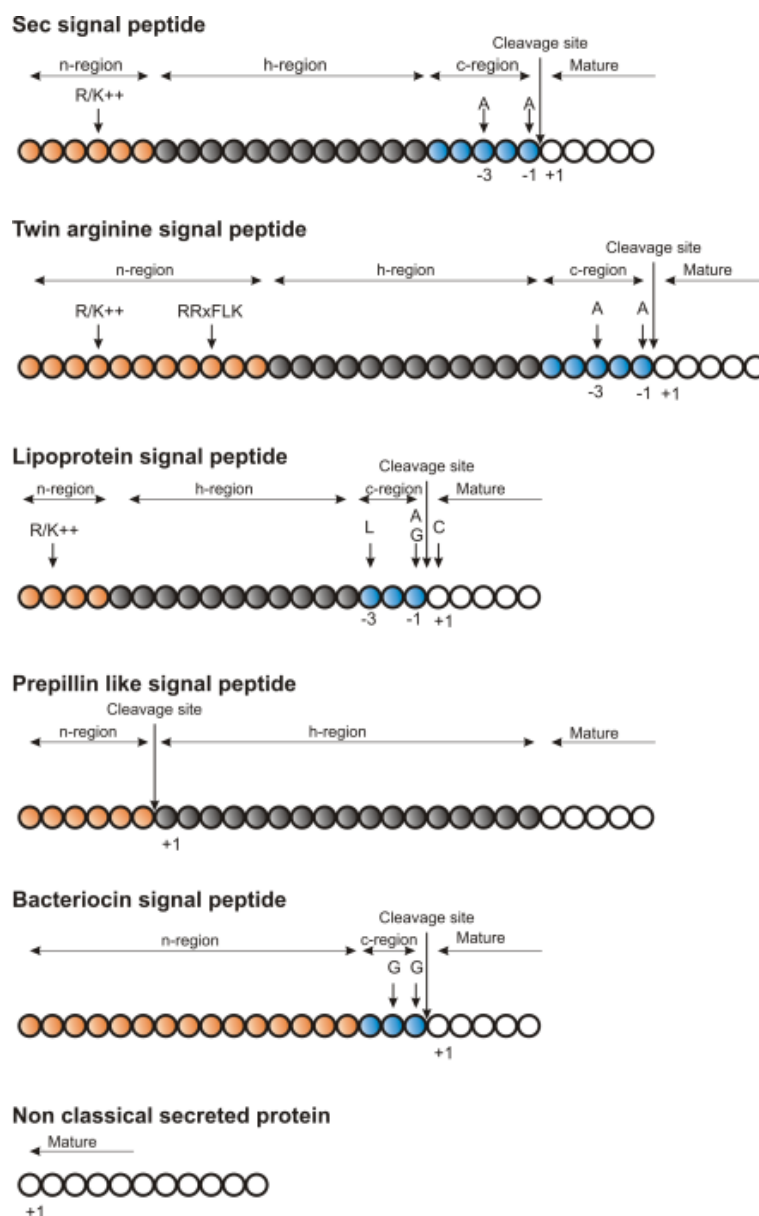
Figure 24.3: *Schematic representation of various signal peptides. Red color indicates n-region, gray color indicates h-region, cyan indicates c-region. All white circles are part of the mature protein.* $+1$ *indicates the first position of the mature protein. The length of the signal peptides is not 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.
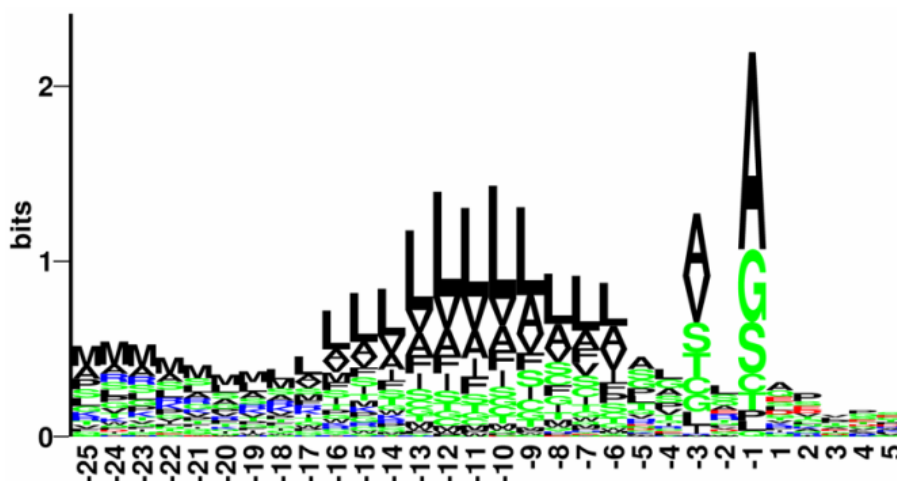
Figure 24.4: *Sequence logo of eukaryotic signal peptides, showing conservation of amino acids in bits [Schneider and Stephens, 1990]. Polar and hydrophobic residues are shown in green and black, respectively, while blue indicates positively charged residues and red negatively charged residues. The logo 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 24.5.

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.
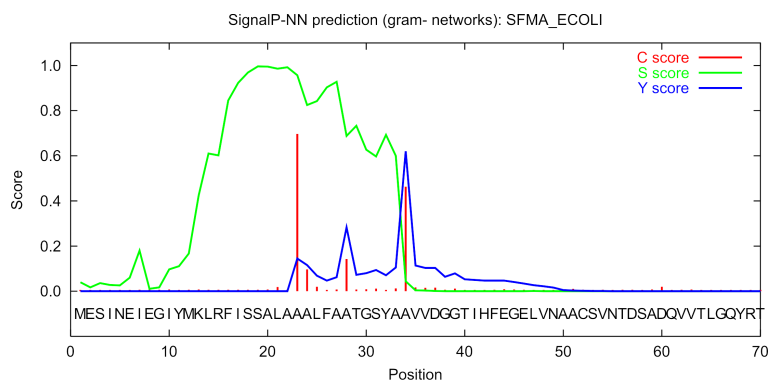
Figure 24.5: *Graphical output from the SignalP method of Swiss-Prot entry* `SFMA_ECOLI`. *Initially this seemed like a borderline prediction, but closer inspection of the sequence revealed an internal methionine at position 12, which could indicate a erroneously annotated start of the protein. Later 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
```

Creative Commons License

All CLC bio's scientific articles are licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. You are free to copy, distribute, display, and use the work for educational purposes, under the following conditions: You must attribute the work in its original form and "CLC bio" has to be clearly labeled as author and provider of the work. You may not use this work for commercial purposes. You may not alter, transform, nor build upon this work.

See `http://creativecommons.org/licenses/by-nc-nd/2.5/` for more information on how to use the contents.

24.2   Protein charge

In *CLC Main 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 Analysis (📁)| Create Protein Charge Plot (📈)**

> or **right-click a protein sequence | Toolbox | Protein Analysis (📁)| Create Protein Charge Plot (📈)**

This opens the dialog displayed in figure 24.6:

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 15.2). If not, click **Finish**.
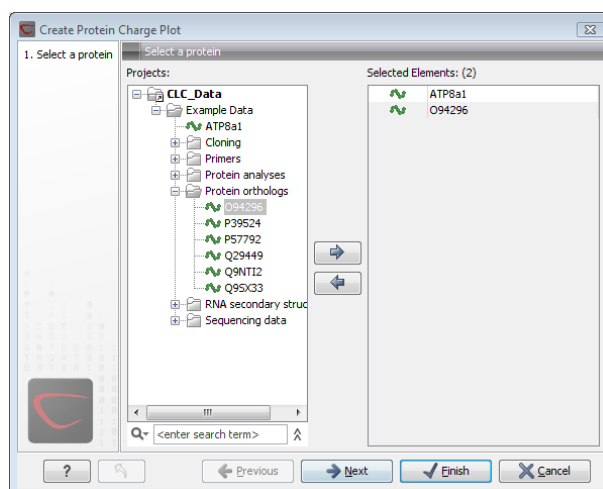
Figure 24.6: *Choosing protein sequences to calculate protein charge.*

24.2.1 Modifying the layout

Figure 24.7 shows the electrical charges for three proteins. In the **Side Panel** to the right, you can modify the layout of the graph.
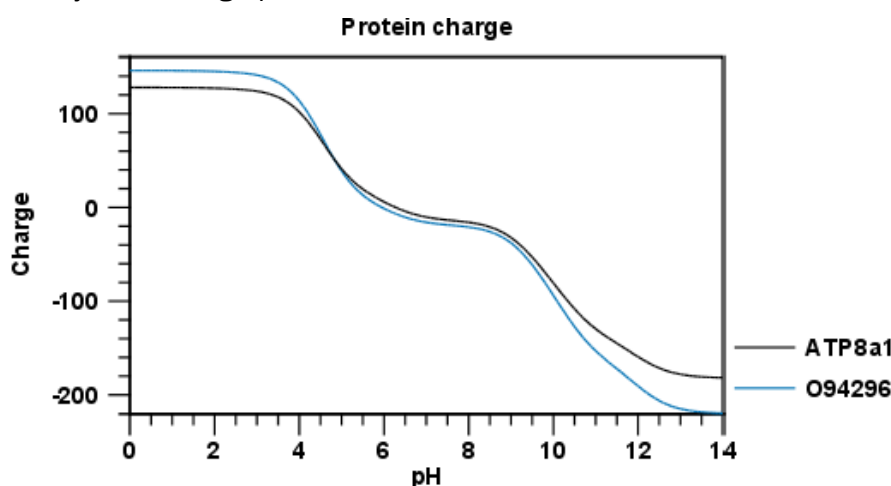
Figure 24.7: *View of the protein charge.*

See section B in the appendix for information about the graph view.

24.3 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 Main 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 plugin* using the plugin manager (see section 1.7.1).

When the plugin is downloaded and installed, you can use it to predict transmembrane helices:

> **Select a protein sequence** | **Toolbox in the Menu Bar** | **Protein Analysis (** **)** | **Transmembrane Helix Prediction (** **)**

> or **right-click a protein sequence** | **Toolbox** | **Protein Analysis (** **)** | **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 15.2). 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 24.8). 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 17.3.3.

Undesired annotations can be removed through the **Delete Annotation (** **)** right-click mouse menu. See section 17.3.5.

24.4 Antigenicity

CLC Main 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 24.8: *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.

24.4.1 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 Analysis ( )| 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 24.9.

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 15.2). If not, click **Finish**. The result can be seen in figure 24.10.

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
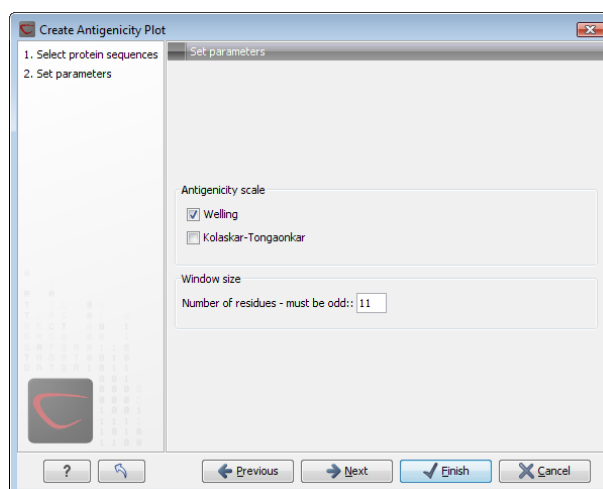
Figure 24.9: *Step two in the Antigenicity Plot allows you to choose different antigenicity scales and the window size.*
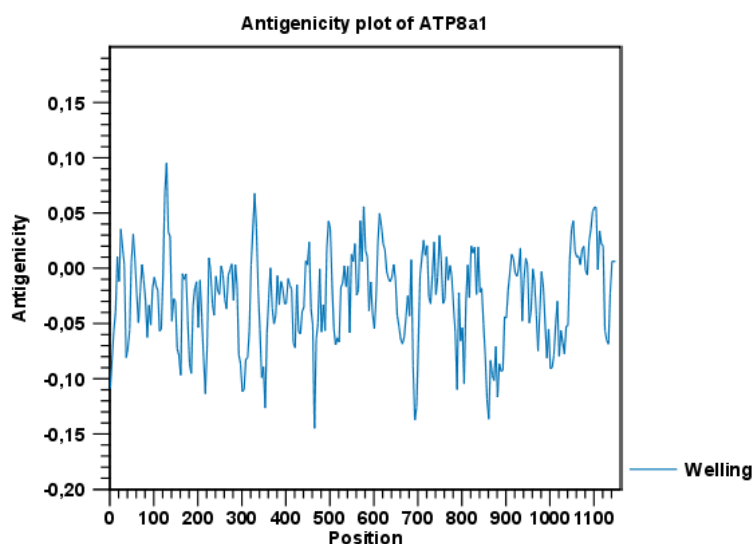
Figure 24.10: *The result of the antigenicity plot calculation and the associated Side Panel.*

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.

24.4.2 Antigenicity graphs along sequence

Antigenicity graphs along the sequence can be displayed using the **Side Panel**. The functionality is similar to hydrophobicity (see section 24.5.2).

24.5 Hydrophobicity

CLC Main Workbench can calculate the hydrophobicity of protein sequences in different ways, using different algorithms. (See section 24.5.3). Furthermore, hydrophobicity of sequences can be displayed as hydrophobicity plots and as graphs along sequences. In addition, *CLC Main Workbench* can calculate hydrophobicity for several sequences at the same time, and for

alignments.

24.5.1   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 Analysis ( )| 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 24.11.
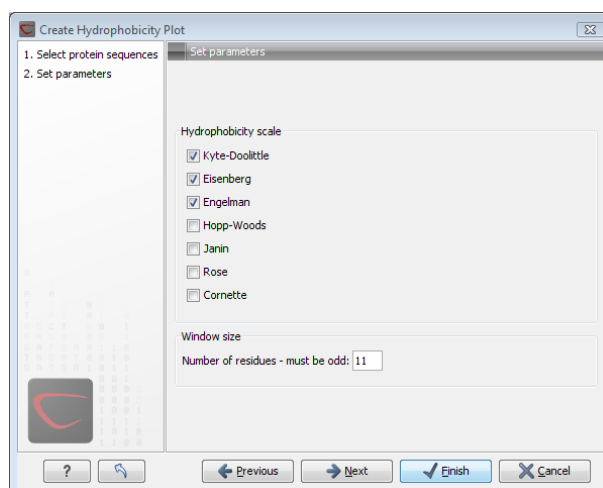
Figure 24.11: *Step two in the Hydrophobicity Plot allows you to choose hydrophobicity scale and the window size.*

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 24.5.3 Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**. The result can be seen in figure 24.12.
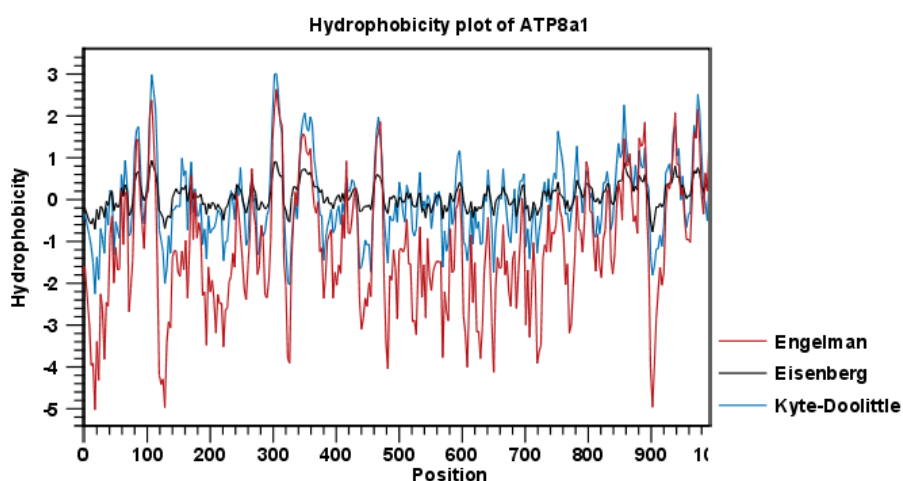
Figure 24.12: *The result of the hydrophobicity plot calculation and the associated Side Panel.*

See section B in the appendix for information about the graph view.

24.5.2 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 24.13.

Figure 24.13: *The different available scales in Protein info in* **CLC Main 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 24.5.3).

In the following we will focus on the different ways that *CLC Main 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 24.14).
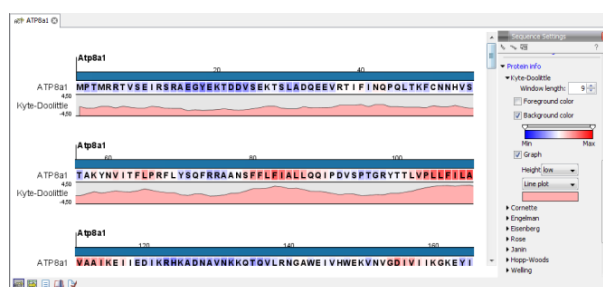
Figure 24.14: *The different ways of displaying the hydrophobicity scores, using the Kyte-Doolittle scale.*

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 24.14. Notice that you can choose the height of the graphs underneath the sequence.

24.5.3 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.
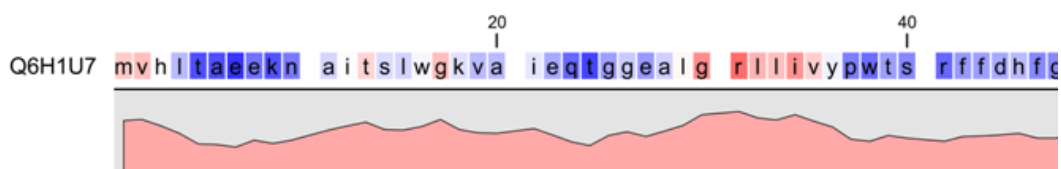
Figure 24.15: *Plot of hydrophobicity along the amino acid sequence. Hydrophobic regions on the sequence have higher numbers according to the graph below the sequence, furthermore hydrophobic regions are colored on the sequence. Red indicates regions with high hydrophobicity and blue indicates regions 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 24.15).

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 us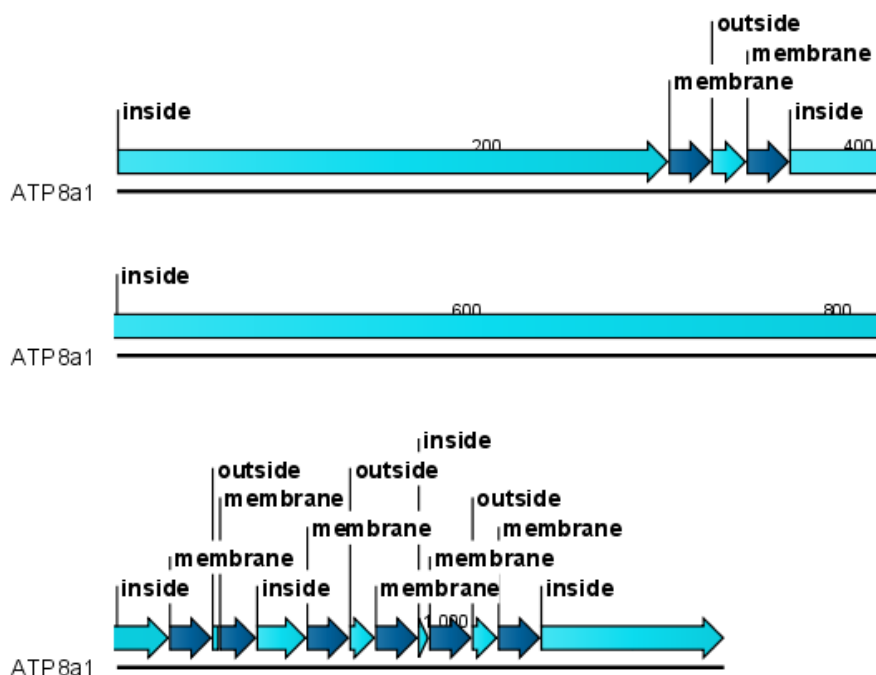eful for prediction of antigenic regions. This method is better than the Hopp-Woods scale of hydrophobicity which is also used to identify antigenic regions.

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.

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 24.1: *Hydrophobicity scales. This table shows seven different hydrophobicity scales which are generally used for prediction of e.g. transmembrane regions and antigenicity.*

24.6   Pfam domain search

With *CLC Main 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 Main 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 17.3.5. Setting a lower cutoff value will result in fewer domains.

In *CLC Main 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 Analysis (⊞)| Pfam Domain Search (⟹)**

or   **right-click a protein sequence | Toolbox | Protein Analysis (⊞)| Pfam Domain Search (⟹)**
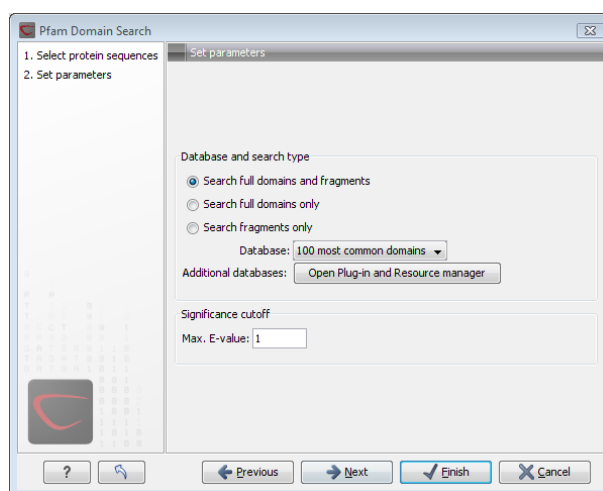
Figure 24.16: *Setting parameters for Pfam domain search.*

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 24.16).

24.6.1 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 Main 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 Main Workbench*, but additional databases can be downloaded and installed as described in section 24.6.2.

- **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 15.2). If not, click **Finish**. This will open a view showing the found domains as annotations on the original sequence (see figure 24.17). If you have selected several sequences, a corresponding number of views will be opened.
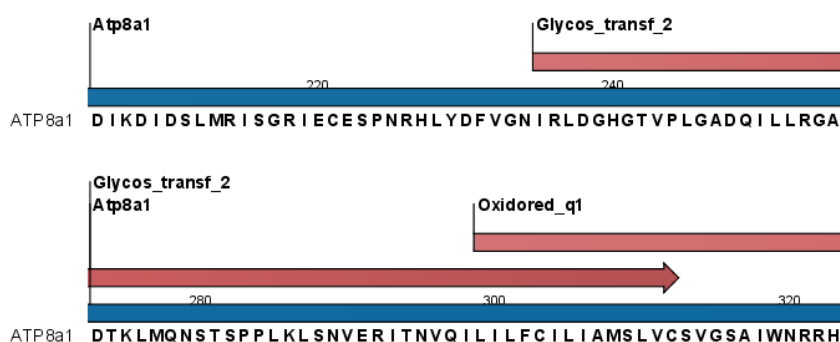
Figure 24.17: *Domains annotations based on Pfam.*

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 tool tip look at `http://pfam.sanger.ac.uk/help#tabview=tab5`.

24.6.2 Download and installation of additional Pfam databases

Additional databases can be downloaded as a resource using the **Plugin manager** (⬚) (see section 1.7.4).

If you are not able to download directly from the Plugin manager, please go to `http://www.clcbio.com/download` to download and install the files directly.

24.7 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 rod like 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 Main 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 Markov 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 Analysis ()| Predict secondary structure ()**

> or **right-click a protein sequence | Toolbox | Protein Analysis ()| Predict secondary structure ()**

This opens the dialog displayed in figure 24.18:
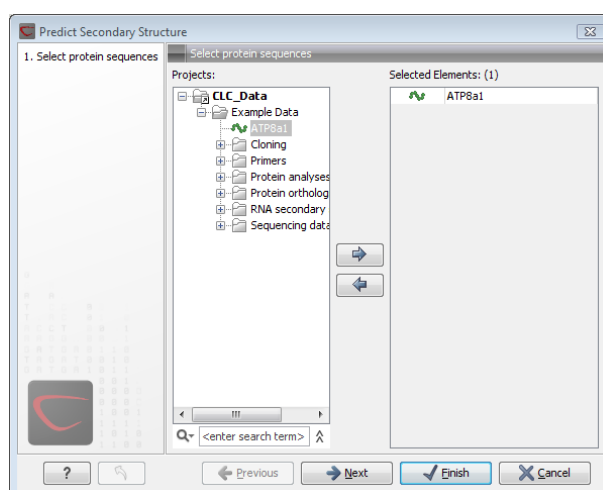
Figure 24.18: *Choosing one or more protein sequences for secondary structure prediction.*

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 15.2). 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 24.19).

Each annotation will carry a tooltip note saying that the corresponding annotation is predicted with *CLC Main Workbench*. Additional notes can be added through the **Edit Annotation ()** right-click mouse menu. See section 17.3.3.
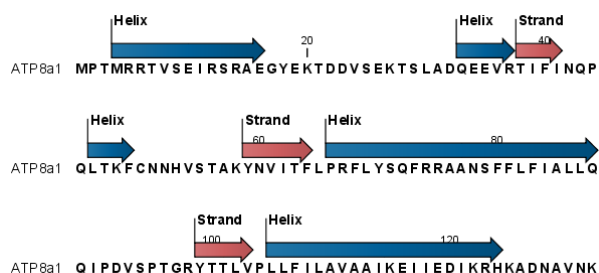
Figure 24.19: *Alpha-helices and beta-strands shown as annotations on the sequence.*

Undesired alpha-helices or beta-sheets can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 17.3.5.

24.8 Protein report

CLC Main 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 Analysis ( )**| **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 22.5 for more about this topic.

- **Plot of charge as function of pH.** See section 24.2 for more about this topic.

- **Plot of hydrophobicity.** See section 24.5 for more about this topic.

- **Plot of local complexity.** See section 22.4 for more about this topic.

- **Dot plot against self.** See section 22.3 for more about this topic.

- **Secondary structure prediction.** See section 24.7 for more about this topic.

- **Pfam domain search.** See section 24.6 for more about this topic.

- **Local BLAST.** See section 19.1.3 for more about this topic.

- **NCBI BLAST.** See section 19.1.1 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 Pfam domain search:

- **Database and search type.** Lets you choose different databases and specify the search for full domains or fragments. See section 24.6.1 for more info about this topic.

- **Significance cutoff.** Lets you set your E-value. See section 24.6.1 for more info about this topic.

In **Step 8** 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.

24.8.1   Protein report output

An example of Protein report can be seen in figure 24.20.

By double clicking a graph in the output, this graph is shown in a different view (*CLC Main Workbench* generates another tab). The report output and the new graph views can be saved by dragging the tab into the **Navigation Area**.

Figure 24.20: *A protein report. There is a Table of Contents in the Side Panel that makes it easy to browse the report.*

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.

24.9   Reverse translation from protein into DNA

A protein sequence can be back-translated into DNA using *CLC Main 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. For background information see section 24.9.2.

In order to make a reverse translation:

> **Select a protein sequence** | **Toolbox in the Menu Bar** | **Protein Analysis (**🔬**)**|
> **Reverse Translate (**〰**)**

>   or   **right-click a protein sequence** | **Toolbox** | **Protein Analysis (**🔬**)**| **Reverse translate**
> **(**〰**)**

This opens the dialog displayed in figure 24.21:

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.
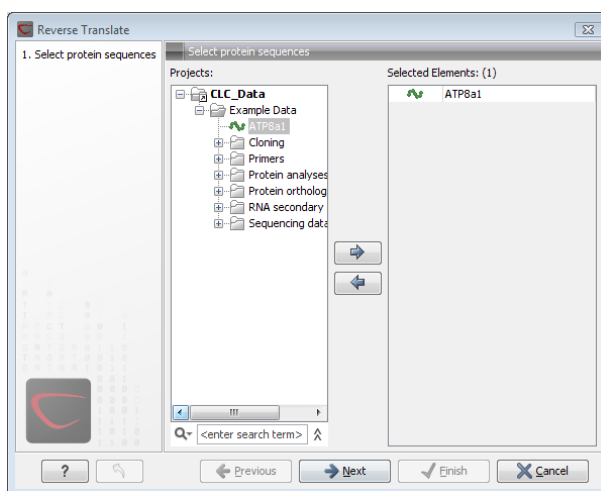
Figure 24.21: *Choosing a protein sequence for reverse translation.*

24.9.1   Reverse translation parameters

Figure 24.22 shows the choices for making the translation.
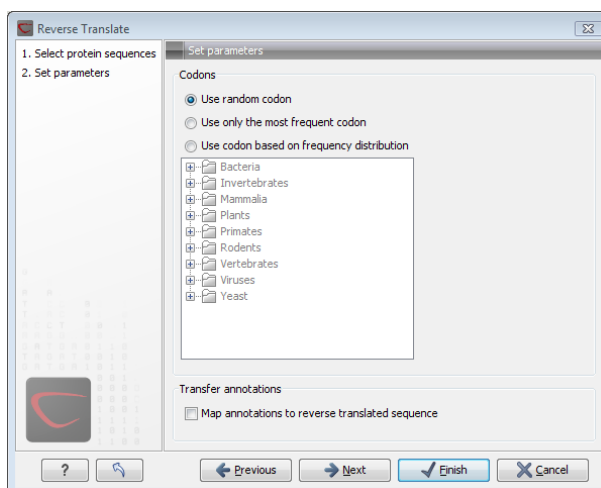
Figure 24.22: *Choosing parameters for the reverse translation.*

- **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. The tables provided were made using Codon Usage database http://www.kazusa.or.jp/codon/ that was built on The NCBI-GenBank Flat File Release 160.0 [June 15 2007]. You can customize the list of codon frequency tables for your installation, see Appendix K.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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.

24.9.2 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

Solving the ambiguities 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.

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

All CLC bio's scientific articles are licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. You are free to copy, distribute, display, and use the work for educational purposes, under the following conditions: You must attribute the work in its original form and "CLC bio" has to be clearly labeled as author and provider of the work. You may not use this work for commercial purposes. You may not alter, transform, nor build upon this work.

See http://creativecommons.org/licenses/by-nc-nd/2.5/ for more information on how to use the contents.

24.10 Proteolytic cleavage detection

CLC Main 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.

24.10.1 Proteolytic cleavage parameters

Given a protein sequence, *CLC Main 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 Analysis (**📷**)| Proteolytic Cleavage, (**✂**)**

This opens the dialog shown in figure 24.23:

CLC Main 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 24.24. The full list of enzymes and their cleavage patterns can be seen in Appendix, section D.
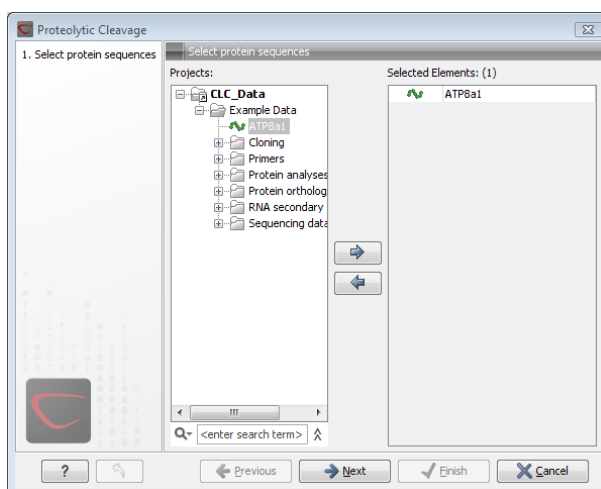
Figure 24.23: *Choosing sequence CAA32220 for proteolytic cleavage.*
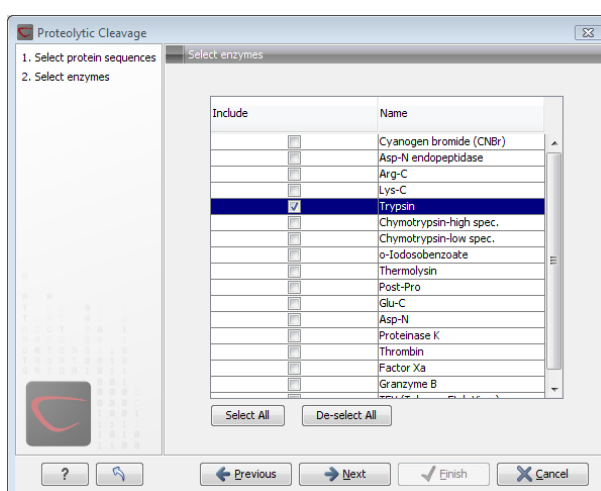
Figure 24.24: *Setting parameters for proteolytic cleavage detection.*

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 24.25 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.

- **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
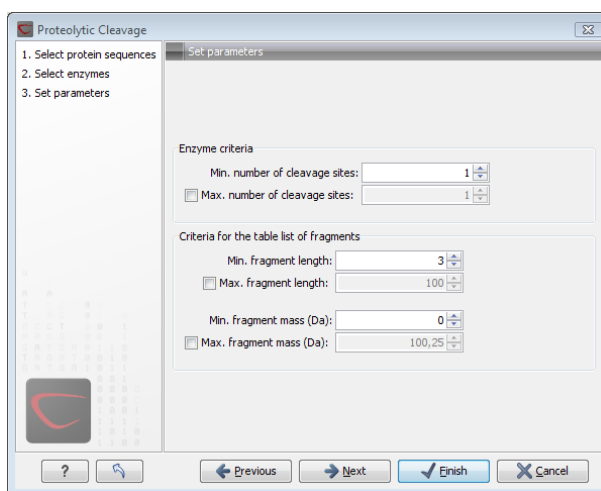
Figure 24.25: *Setting parameters for proteolytic cleavage detection.*

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 15.2). If not, click **Finish**.

The result of the detection is displayed in figure 24.26.
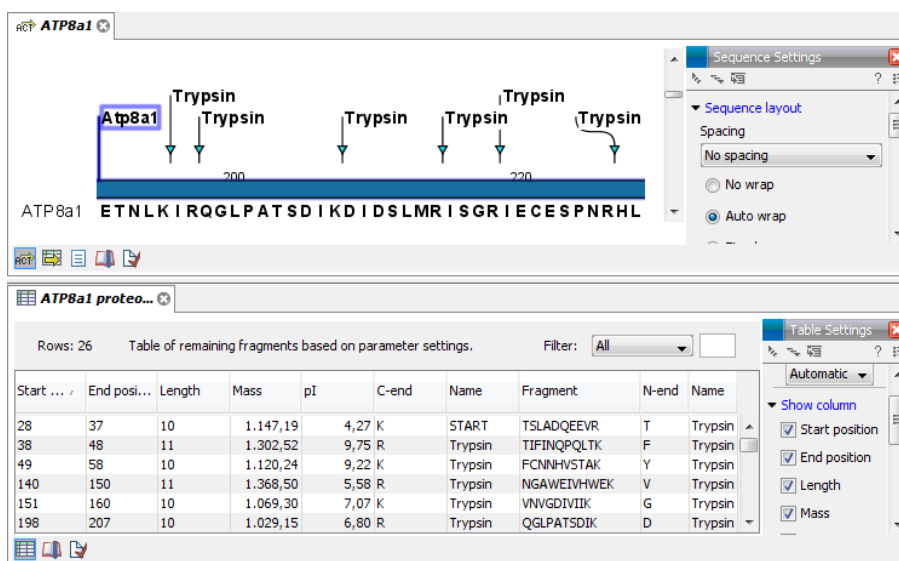
Figure 24.26: *The result of the proteolytic cleavage detection.*

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.

24.10.2 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 24.27.

Cleavage site

P4 – P3 – P2 – P1 ┆ P1' – P2' – P3'
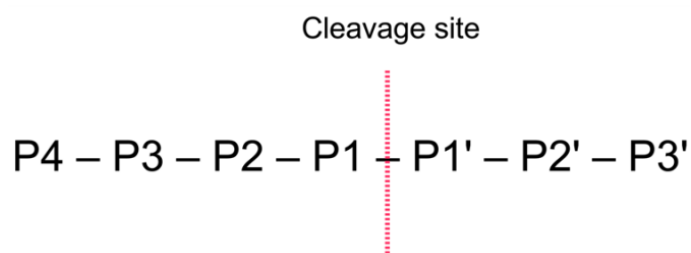
Figure 24.27: *Nomenclature of the peptide substrate. The substrate is cleaved between position P1-P1'.*

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 24.28).
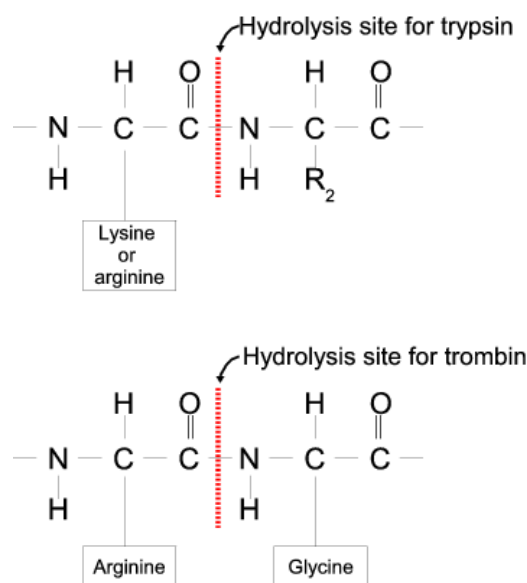
Figure 24.28: *Hydrolysis of the peptide bond between two amino acids. Trypsin cleaves unspecifically at lysine or arginine residues whereas trombin cleaves at arginines if asparate or glutamate is absent.*

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 25

Primers

Contents

CLC Main 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.

25.1 Primer design - an introduction

Primer design can be accessed in two ways:

> **select sequence** | **Toolbox in the Menu Bar** | **Primers and Probes (**![icon]**)**| **Design Primers (**![icon]**)** | **OK**

> or **right-click sequence** | **Show** | **Primer (**![icon]**)**

In the primer view (see figure 25.1), the basic options for viewing the template sequence are the same as for the standard sequence view. See section 17.1 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.
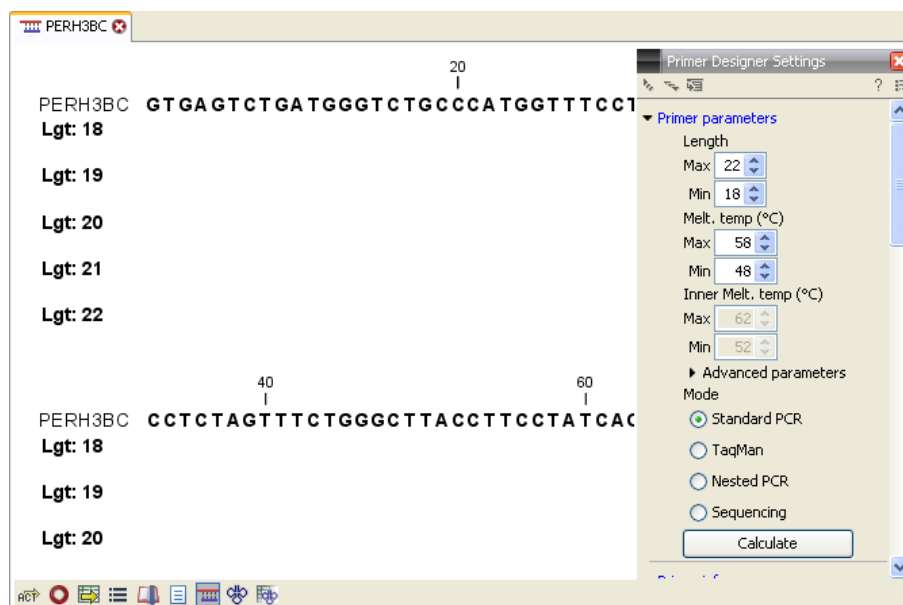
Figure 25.1: *The initial view of the sequence used for primer design.*

25.1.1 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 25.2).
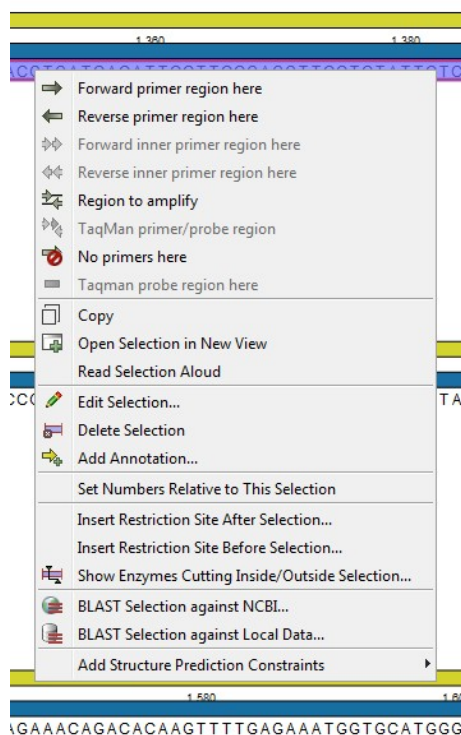
Figure 25.2: *Right-click menu allowing you to specify regions for the primer design*

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.

25.1.2   Scoring primers

CLC Main 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.

25.2   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 25.3).

Figure 25.3: *The two groups of primer parameters (in the program, the Primer information group is listed below the other group).*

25.2.1 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 combined concentration of all 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

25.3   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 25.3).

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.

25.3.1   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 25.4).
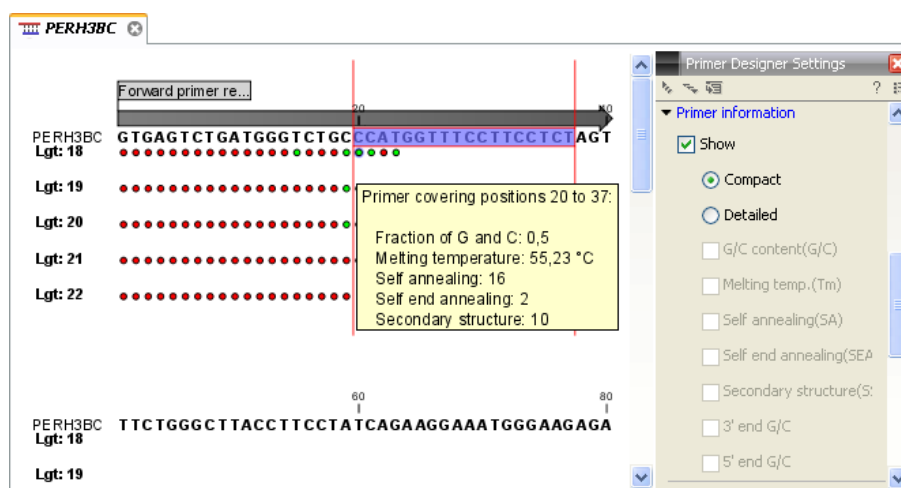
Figure 25.4: *Compact information mode*

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.

25.3.2   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 25.5).
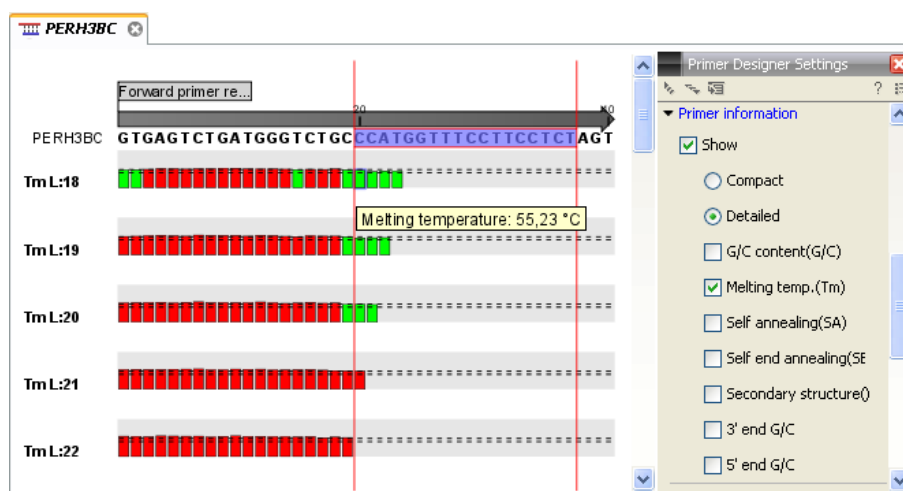
Figure 25.5: *Detailed information mode*

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.

25.4 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 25.6).

Figure 25.6: *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.

25.4.1   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.

25.4.2   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.

25.4.3   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**.

25.5   Standard PCR

This mode is used to design primers for a PCR amplification of a single DNA fragment.

25.5.1 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 25.3) 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 25.7).
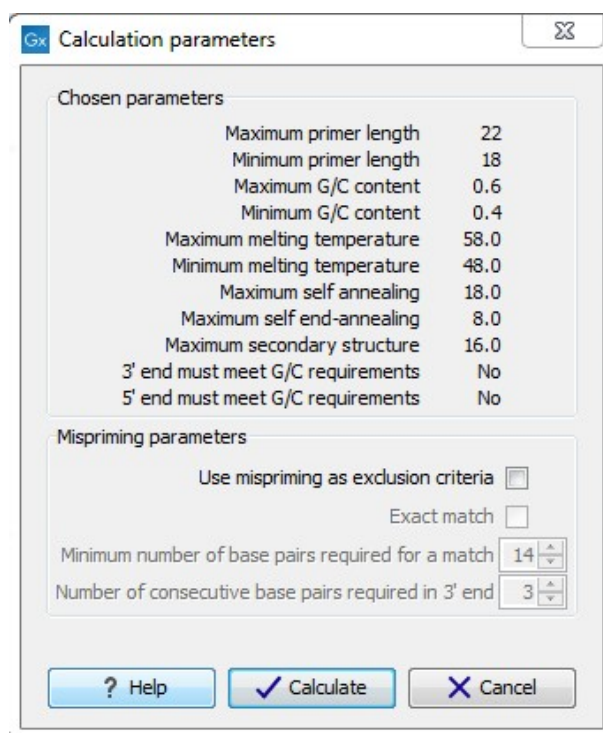
Figure 25.7: *Calculation dialog for PCR primers when only a single primer region has been defined.*

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 25.8).

Figure 25.8: *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.

25.5.2 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. The scale is from 0 to 100.

- 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

25.6   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 25.3) 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 25.9).

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:

Figure 25.9: *Calculation dialog*

- Maximum percentage point difference in G/C content (described above under Standard PCR) - this criteria is applied to both primer pairs independently.

- 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.

25.6.1 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.

25.7 TaqMan

CLC Main 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.

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 25.3) 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 25.10) which is similar to the *Nested PCR* dialog described above (see section 25.6).

Figure 25.10: *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.

25.7.1 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.

25.8 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 25.3) 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 25.11).
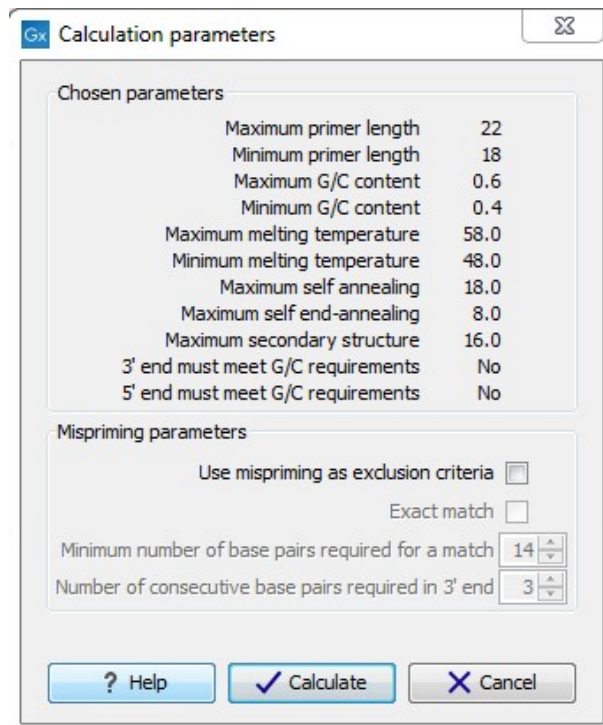
Figure 25.11: *Calculation dialog for sequencing primers*

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 25.5 for a description.

25.8.1   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.

25.9   Alignment-based primer and probe design

CLC Main 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 25.12), the basic options for viewing the template alignment are the same as for the standard view of alignments. See section 29 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.

Figure 25.12: *The initial view of an alignment used for primer design.*

25.9.1 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 left 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 25.12):

- 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.

25.9.2 Alignment based design of PCR primers

In this mode, a single or a pair of PCR primers are designed. *CLC Main 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 hetero-geneities 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 25.13.

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.
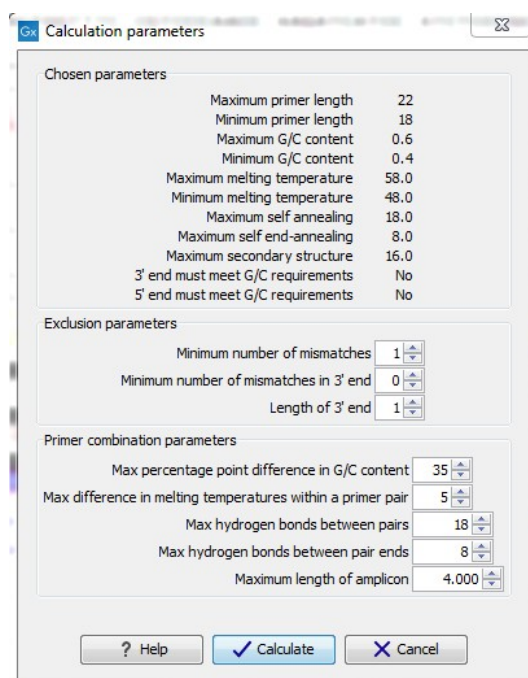
Figure 25.13: *Calculation dialog shown when designing alignment based PCR primers.*

25.9.3 Alignment-based TaqMan probe design

CLC Main 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 25.14 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.

25.10 Analyze primer properties

CLC Main 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 ( )
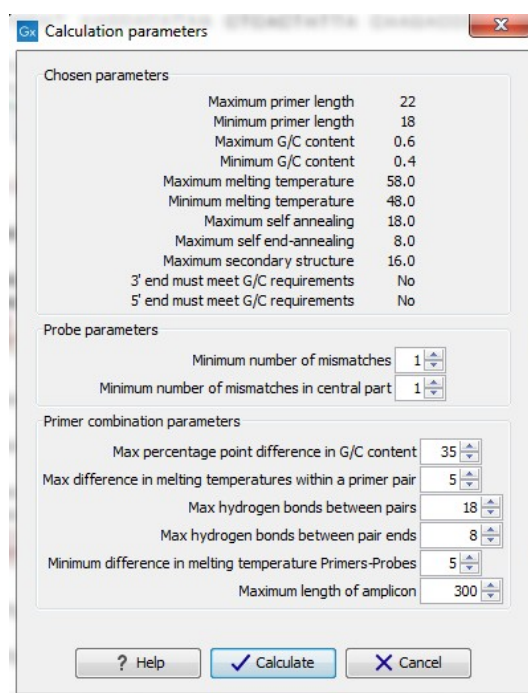
Figure 25.14: *Calculation dialog shown when designing alignment based TaqMan probes.*

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 25.15:
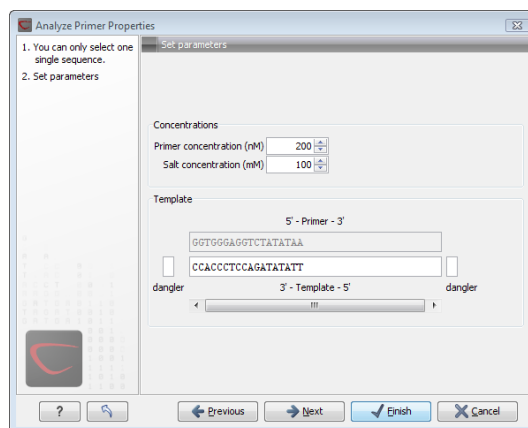
Figure 25.15: *The parameters for analyzing primer properties.*

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 15.2). If not, click **Finish**. The result is shown in figure 25.16:
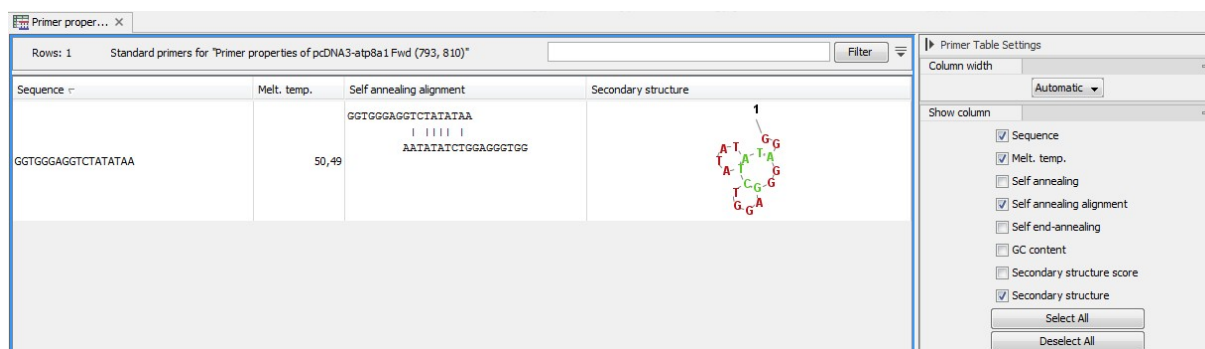
Figure 25.16: *Properties of a primer from the Example Data.*

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 25.5.2.

25.11 Find binding sites and create fragments

In *CLC Main 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.

25.11.1 Binding parameters

This opens the dialog displayed in figure 25.17:

At the top, select one or more primers by clicking the browse (🔬) button. In *CLC Main Workbench*, primers are just DNA sequences like any other, but there is a filter on the length of the sequence.
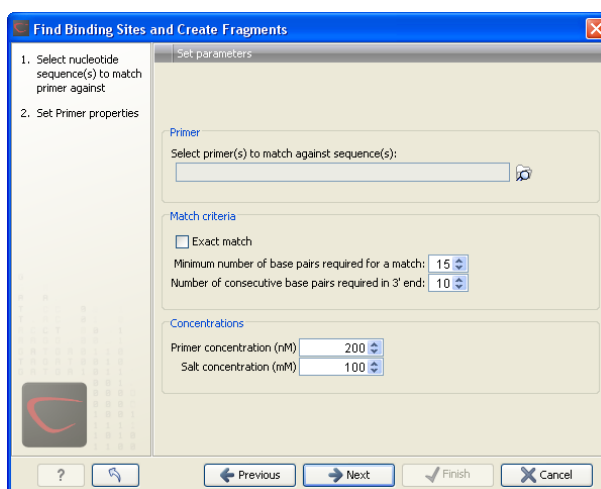
Figure 25.17: *Search parameters for finding primer binding sites.*

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)

25.11.2 Results - binding sites and fragments

Click **Next** to specify the output options as shown in figure 25.18:

The output options are:

- **Add binding site annotations**. This will add annotations to the input sequences (see details below).

- **Create binding site table**. Creates a table of all binding sites. Described in details below.
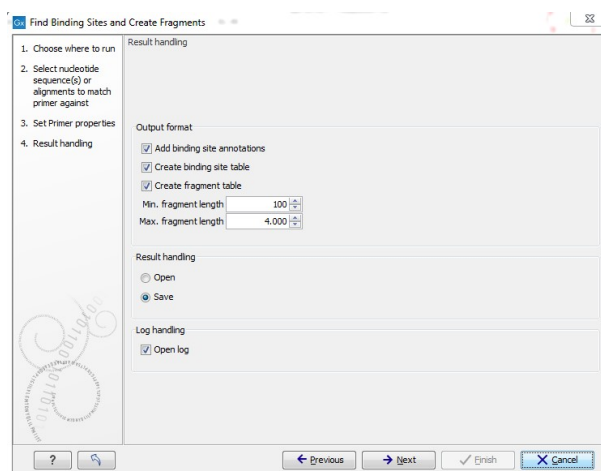
Figure 25.18: *Output options include reporting of binding sites and fragments.*

- **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 15.2). If not, click **Finish**.

An example of a **binding site annotation** is shown in figure 25.19.
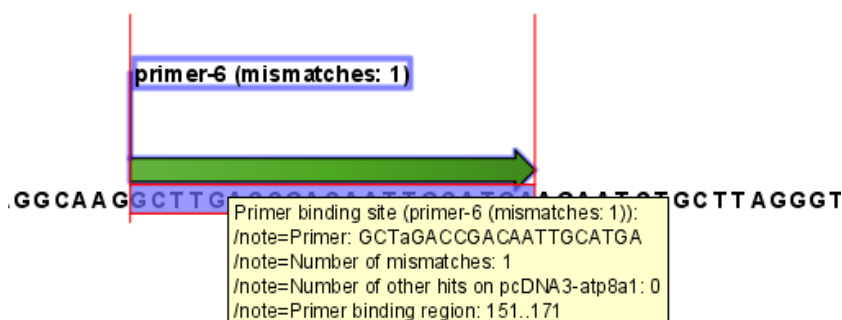
Figure 25.19: *Annotation showing a primer match.*

The annotation has the following information:

- **Sequence of the primer**. Positions with mismatches will be in lower-case (see the fourth position in figure 25.19 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.

An example of the **primer binding site table** is shown in figure 25.20.

Figure 25.20: *A table showing all binding sites.*

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 25.5.2. You can use this table to browse the binding sites. If you make a split view of the table and the sequence (see section 9.1.6), 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 25.21.

Figure 25.21: *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 9.1.6), 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 25.22.

Figure 25.22: *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 25.22, 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 28.1).
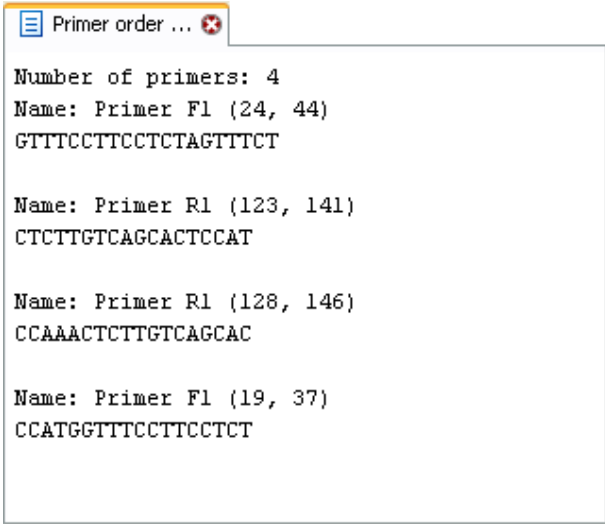
25.12   Order primers

To facilitate the ordering of primers and probes, *CLC Main 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 25.23). 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 25.23

Figure 25.23: *A primer order for 4 primers.*

Chapter 26

Sequencing data analyses and Assembly

Contents

CLC Main Workbench lets you import, trim and assemble DNA sequence reads from automated sequencing machines. A number of different formats are supported (see section 13.1). 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.

26.1 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 13.1).

After import, the sequence reads and their trace data are saved as DNA sequences. This means that all analyses that 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 26.1.
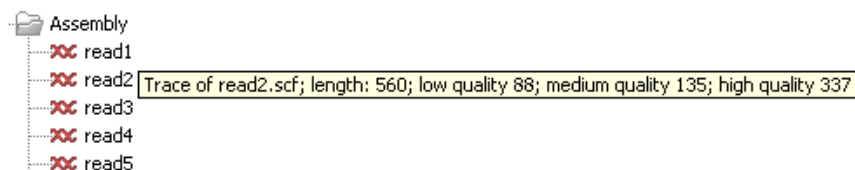
Figure 26.1: *A tooltip displaying information about the quality of the chromatogram.*

The qualities are based on the phred scoring system, with scores below 19 counted as low quality, scores between 20 and 39 counted as medium quality, and those 40 and above counted as high quality.

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 ( ).

26.1.1 Scaling traces

The traces can be scaled by dragging the trace vertically as shown in figure figure 26.2. 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 26.1.2.
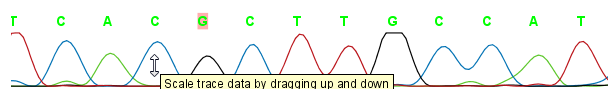
Figure 26.2: *Grab the traces to scale.*

26.1.2 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 26.3):

- **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 26.1.1.

Figure 26.3: *A sequence with trace data. The preferences for viewing the trace are shown in the Side Panel.*

When working with stand along mappings containing reads with trace data, you can view the traces by turning on the trace setting options as described here **and** choosing **Not compact** in the Read layout setting for the mapping. Please see section 26.8.1.

26.2 Demultiplex reads

Multiplexing techniques are often used When sequencing of different samples in one sequencing run. One method used is 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 read will have a sample-specific tag, which is a specific sequence of nucleotides before and after the sequence of interest. This principle is shown in figure 26.4 (please refer to [Meyer et al., 2007] for more detailed information).

The sample-specific tag - also called the barcode - can then be used to distinguish between the different samples when analyzing the sequence data.

The post-processing of the sequencing data to separate the reads into their corresponding samples based on their barcodes, can be done using the demultiplexing functionality of the *CLC Main Workbench*. Using this tool, sequences are associated with a particular samples when they contain an exact match to a particular barcode. Sequences that do not contain an exct match to any of the barcode sequences provided are classfied as not grouped and are put into a sequence list with the name "Not grouped".

Note that there is also an example using Illumina data at the end of this section.

Please note that demultiplexing is often carried out on the sequencing machine so that the sequencing reads are already separated according to sample. This is often the best option, if it
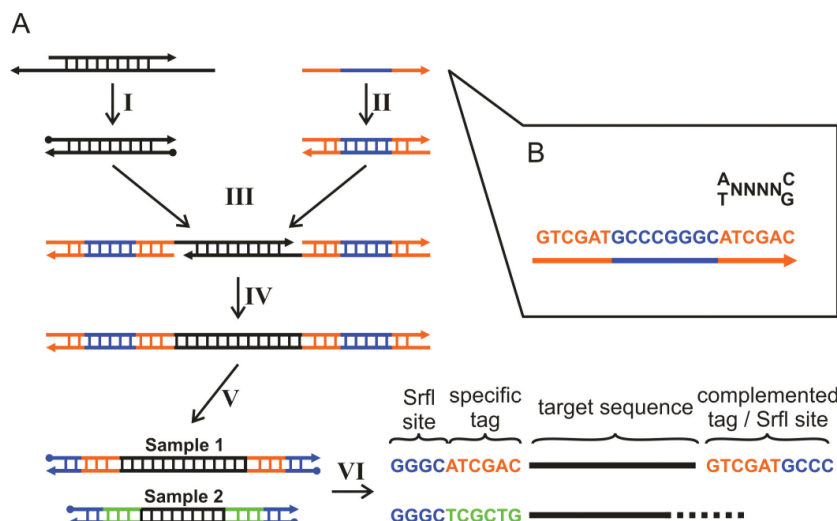
Figure 26.4: *Tagging the target sequence. Figure from [Meyer et al., 2007].*

is available to you. Of course, in such cases, the data will not need to be demuliplexed again after import into the *CLC Main Workbench*.

To de-multiplex your data, please go to:

Toolbox | NGS Core Tools ( ) | Demultiplex Reads ( )

This opens a dialog where you can specify the sequences to process.

When you click on the button labeled **Next**, you can then specify the details of how the demultiplexing 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 26.5.
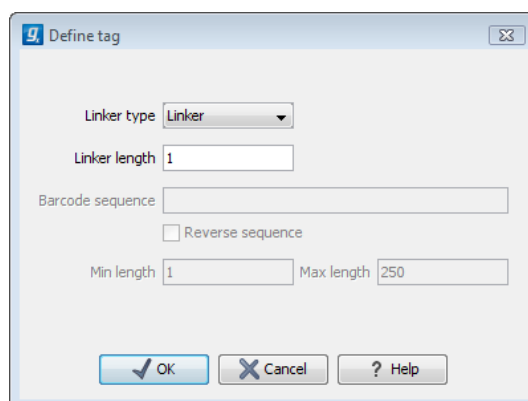
Figure 26.5: *Defining an element of the barcode system.*

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 26.4, 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. In this dialog, you simply need to specify the length of the barcode. The valid sequences for your barcodes need to be provided at a later stage in setting up this job.

- **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 26.4, 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 26.6.
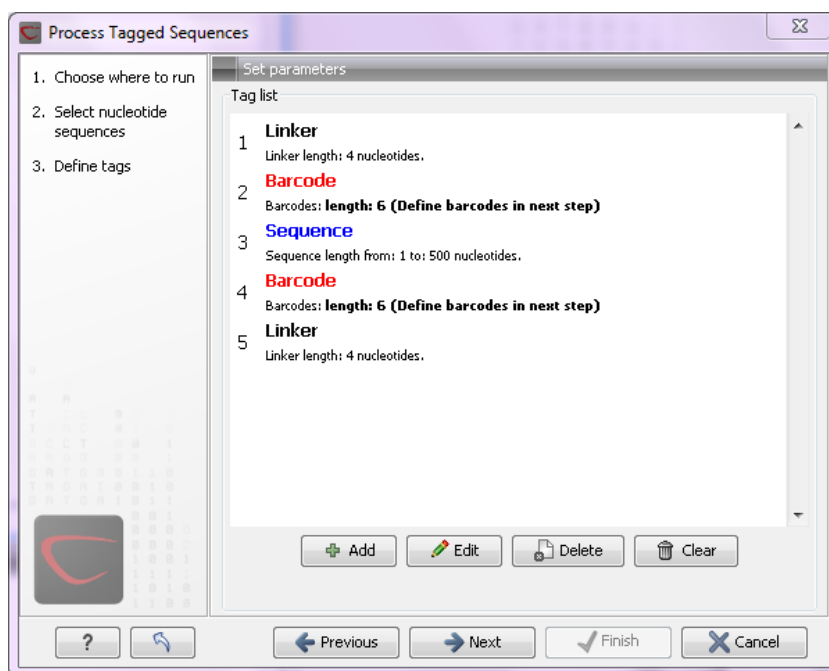
Figure 26.6: *Processing the tags as shown in the example of figure 26.4.*

If you have paired data, the dialog shown in figure 26.6 will be displayed twice - one for each part of the pair.

In case, where paired reads are expected to be barcoded in the same way (see example below), you would set the parameters for read1 (wizard step 3) and read2 (wizard step 4) to be the same.

Read1 : –Linker–Barcode1–Sequence

Read2 : –Linker–Barcode1–Sequence

However, if read2 of the pair is not expected to be the same as read1 in the pair, it is necessary to adjust these settings accordingly. For example, it is possible that read2 does not contain any barcode sequence at all. In this case, you would simply set the sequence parameter for the mate and exclude the barcode and linker parameters.

Clicking **Next** will display a dialog as shown in figure 26.7.

Barcodes can be entered manually or imported from a properly formatted CSV or Excel file:
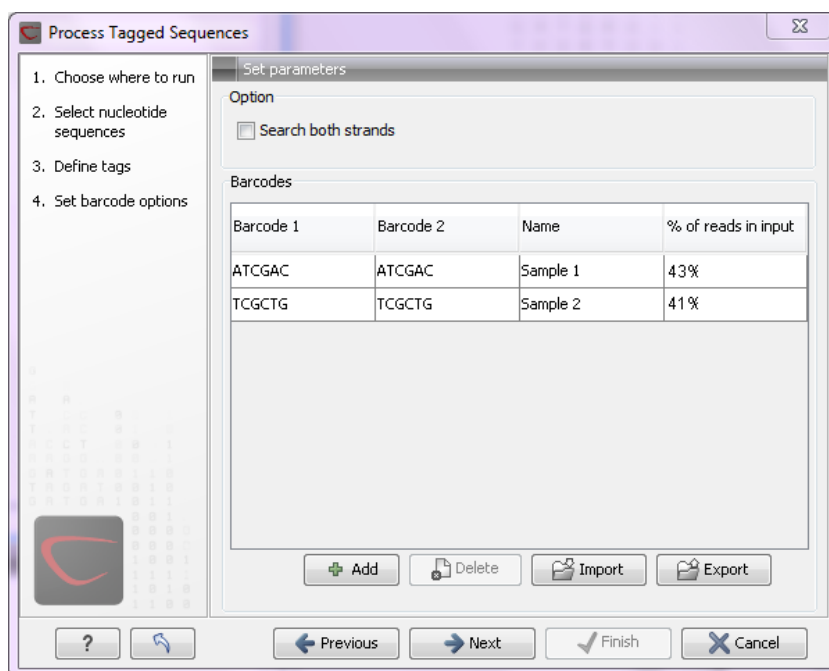
Figure 26.7: *Specifying the barcodes as shown in the example of figure 26.4.*

Manually 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 barcode name is used when naming the results.

Import from CSV or Excel To import a file of barcodes, click on the **Import** (📥) button. The input format consists of two columns: the first contains the barcode sequence, the second contains the name of the barcode. An acceptable csv format file would contain columns of information that looks like:

```
  "AAAAAA","Sample1"
 "GGGGGG","Sample2"
 "CCCCCC","Sample3"
```

The **Preview** column will show a preview of the results by running through the first 10,000 reads.

At the top, you can choose to search on both strands for the barcodes (this is needed for some 454 protocols where the MID is located at either end of the read).

Click **Next** to specify the output options. First, you can choose to create a list of the reads that could not be grouped. Second, you can create a summary report showing how many reads were found for each barcode (see figure 26.8).

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 15.1).

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.

1 Demultiplexing summary

1.1 Reads per barcode

Barcode	Number of reads	Percentage of reads
Barcode:GGT	1,745,043	26%
Barcode:CGT	1,305,703	20%
Barcode:AAT	1,850,050	28%
Barcode:CCT	1,251,849	19%
Not grouped	445,560	7%

1.2 Reads per barcode
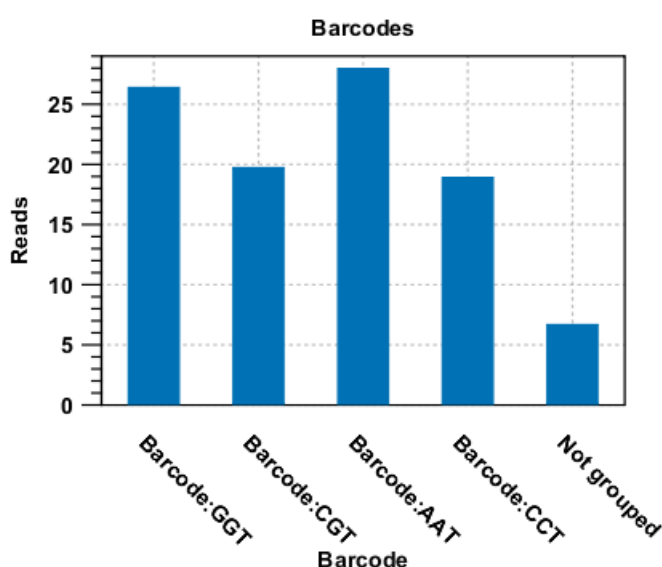
Figure 26.8: *An example of a report showing the number of reads in each group.*

An example using Illumina barcoded sequences

The data set in this example can be found at the Short Read Archive at NCBI: http://www.ncbi.nlm.nih.gov/sra/SRX014012. It can be downloaded directly in fastq format via the URL http://trace.ncbi.nlm.nih.gov/Traces/sra/sra.cgi?cmd=dload&run_list=SRR030730&format=fastq. The file you download can be imported directly into the Workbench.

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 26.9.

Click **Next** to specify the bar codes as shown in figure 26.10 (use the **Add** button).

With this data set we got the four groups as expected (shown in figure 26.11). The **Not grouped** list contains 445,560 reads that will have to be discarded since they do not have any of the
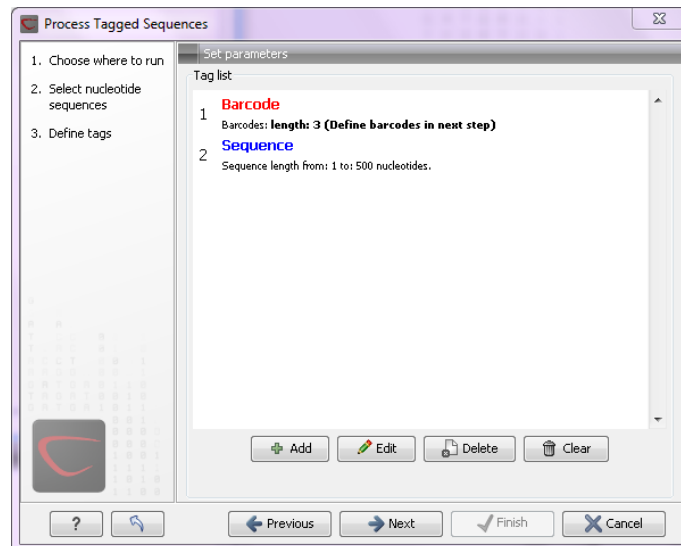
Figure 26.9: *Setting the barcode length at three*
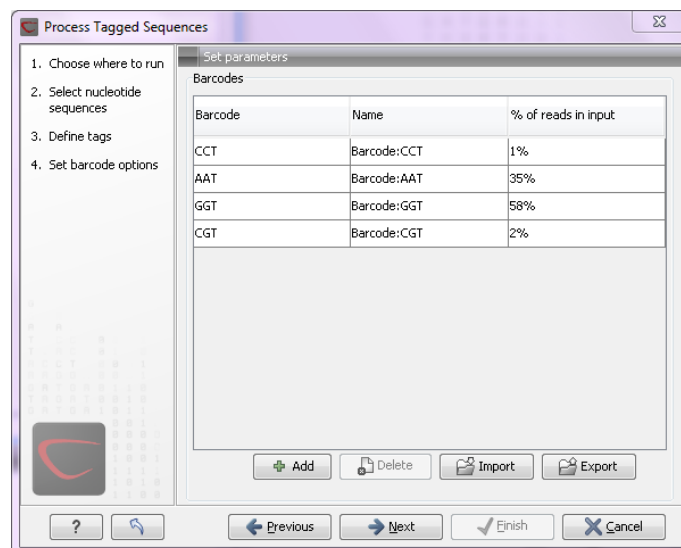
Figure 26.10: *A preview of the result*

barcodes.
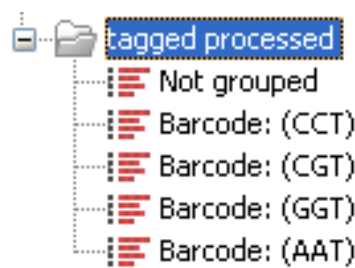
Figure 26.11: *The result is one sequence list per barcode and a list with the remainders*

26.3 Trim sequences

Trimming as described in this section involves marking of low quality and/or vector sequence with a Trim annotation as shown in figure 26.12). Such annotated regions are then ignored when using downstream analysis tools located in the same section of the Workbench toolbox, for example Assembly (see section 26.4). The trimming described here annotates, but does not remove data, allowing you to explore the output of different trimming schemes easily.

Trimming as a separate task can be done manually or using a tool designed specifically for this task.

To remove existing trimming information from a sequence, simply remove its trim annotation (see section 17.3.3).
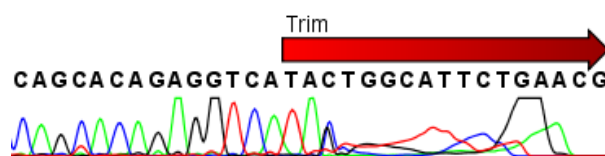
Figure 26.12: *Trimming creates annotations on the regions that will be ignored in the assembly process.*

When exporting sequences in fasta format, there is an option to remove the parts of the sequence covered by trim annotations.

26.3.1 Trimming using the Trim tool

Sequence reads can be trimmed based on a number of different criteria. Using a trimming tool for this is particularly useful if:

- You have many sequences to trim.

- You wish to trim vector contamination from sequencing reads.

- You wish to ensure that consistency when trimming. That is, you wish to ensure the same criteria are used for all the sequences in a set.

To start up the Trim tool in the Workbench, go to the menu option:

  Toolbox | Sequencing Data Analysis (📶**)| Trim Sequences (**🧬**)**

This opens a dialog where you can choose the sequences to trim.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 26.13.

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.
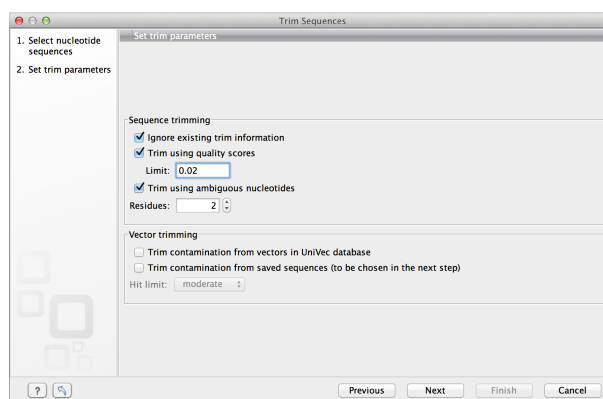
Figure 26.13: *Setting parameters for trimming.*

- **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, and formats using other scales will be converted during import. The Phred quality scores (Q), defined as: $Q = -10log10(P)$, where P is the base-calling error probability, can then be used to calculate the error probabilities, which in turn can be used to set the limit for, which bases should be trimmed.

  Hence, the first step in the trim process is to convert the quality score (Q) to an 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 not trimmed will be 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.

  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 mark sequence ends with significant matches with a 'Trim' annotation (the database is included when you install the *CLC Main 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 Main 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 ≥ 24.
    · Internal match with Score ≥ 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 have imported into the Workbench. 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 15.2). 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). Note that the trim annotations are used to signal that this part of the sequence is to be ignored during further analyses, hence the trimmed sequences are not deleted. If there are no trim annotations, the sequence has not been trimmed.

26.3.2 Manual trimming

Sequence reads can be trimmed manually while inspecting their trace and quality data.

Trimming sequences manually involves adding an annotation of type Trim, with the special condition that this annotation 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 a trimming annotation to the end of the sequence in the selected direction. No sequence is being deleted here. Rather, the regions covered by trim annotations are noted by downstream analyses (in the same section of the Workbench Toolbox as the Trim tool) as regions to be ignored.

26.4   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 26.6). To perform the assembly:

> **select sequences to assemble** | **Toolbox in the Menu Bar** | **Sequencing Data Analysis** ( ) | **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, you need the *CLC Genomics Workbench* (see `http://www.clcbio.com/genomics`).

When the sequences are selected, click **Next**. This will show the dialog in figure 26.14
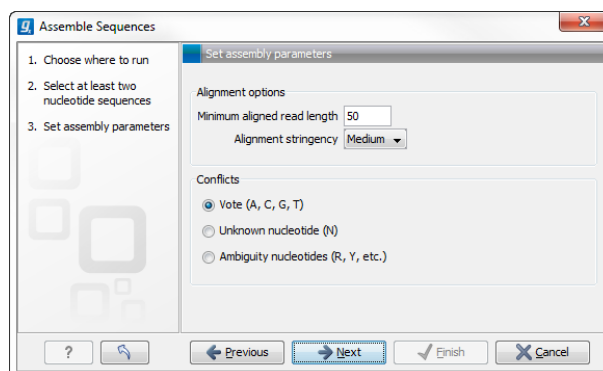
Figure 26.14: *Setting assembly parameters.*

This dialog gives you the following options for assembling:

- **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 (conflicts are registered already when two nucleotides differ).

- **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the reads (nucleotide ambiguity is registered already when two nucleotides differ). For an overview of ambiguity codes, see Appendix H.

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 26.8.4.

- **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 26.8.

- **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 26.8.7.

- **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.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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 26.8 on how to use the resulting contigs.

26.5   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 date 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 | Molecular Biology Tools ( ) | 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 date and analysis position, these two parts should be checked as shown in figure 26.15.
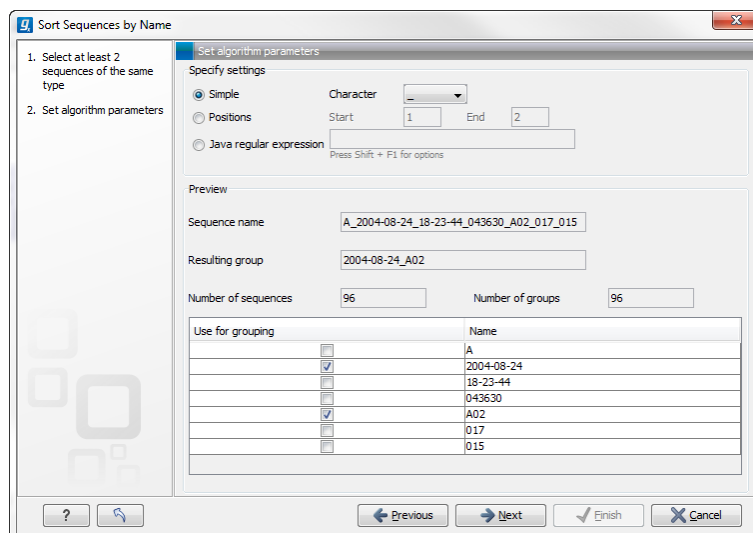
Figure 26.15: *Splitting up the name at every underscore (_) and using the date and analysis position for grouping.*

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 15.2). If not, click **Finish**. A new sequence list will be generated for each group. It will be named according to the group, e.g. *2004-08-24_A02* will be the name of one of the groups in the example shown in figure 26.15.

Advanced splitting using regular expressions

You can see a more detail explanation of the regular expressions syntax in section 22.8.3. 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 26.15 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 26.16. The round brackets () denote the part of the name that will be listed in the groups table at the
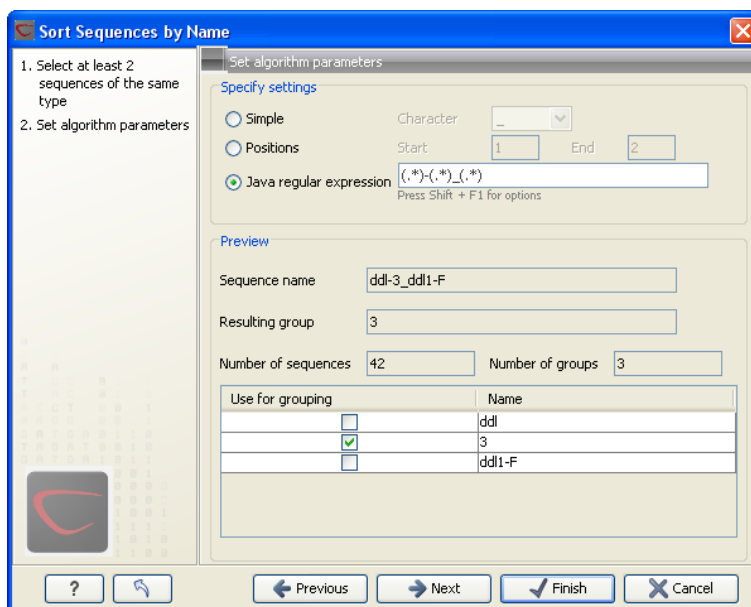
Figure 26.16: *Dividing the sequence into three groups based on the number in the middle of the name.*

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.

26.6 Assemble sequences to reference

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 Analysis (⚑)| Assemble Sequences to Reference (⚑)**

This opens a dialog where you can alter your choice of sequences that you wish to assemble. You can also add sequence lists.

Note! You can assemble a maximum of 2000 sequences at a time.

To assemble more sequences, you need the *CLC Genomics Workbench* (see http://www. clcbio.com/genomics).

When the sequences are selected, click **Next**, and you will see the dialog shown in figure 26.17
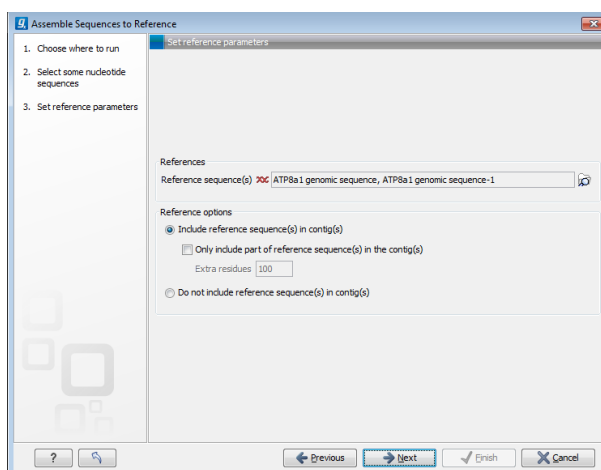
Figure 26.17: *Parameters for how the reference should be handled when assembling sequences to a reference sequence.*

This dialog gives you the following options for assembling:

- **Reference sequence.** Click the **Browse and select element** icon (🔍) in order to select one or more sequences to use as reference(s).

- **Include reference sequence(s) in contig(s).** This will create a contig for each reference with the corresponding reference sequence at the top and the aligned sequences 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 reference sequence(s) in the contig(s).** If the aligned sequences only cover a small part of a reference sequence, it may not be desirable to include the whole reference sequence in a contig. When this option is selected, you can specify the number of residues from reference sequences that should be included on each side of regions spanned by aligned sequences using the **Extra residues** field.

- **Do not include reference sequence(s) in contig(s).** This will produce contigs without any reference sequence where the input sequences have been assembled using reference sequences as a scaffold. The input sequences are first aligned to the reference sequence(s). Next, the consensus sequence for regions spanned by aligned sequences are extracted and output as contigs. This option is useful when performing assembling sequences where the reference sequences that are not closely related to the input sequencing.

When the reference sequence has been selected, click **Next**, to see the dialog shown in figure 26.18
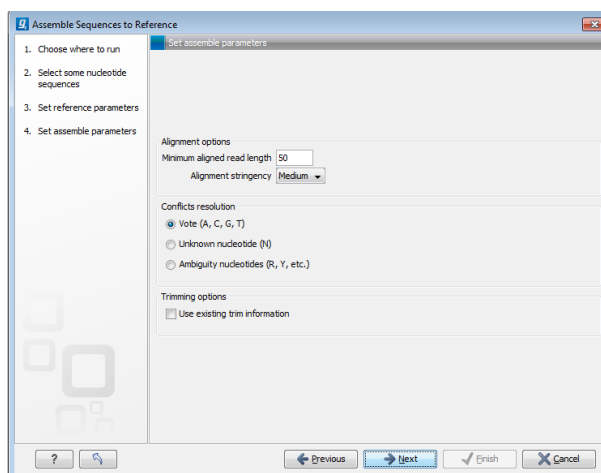
Figure 26.18: *Options for how the input sequences should be aligned and how nucleotide conflicts should be handled.*

In this dialog, you can specify the following options:

- **Minimum aligned read length.** The minimum number of nucleotides in a read which must match a reference sequence. If an input sequence does not meet this criteria, the sequence is excluded from the assembly.

- **Alignment stringency.** Specifies the stringency of the scoring function used for aligning the input sequences to the reference sequence(s). A higher stringency level often produce contigs with lower levels of ambiguity but also reduces the ability to align distant homologs or sequences with a high error rate to reference sequences. The result of a higher stringency level is often that the number of contigs increases and the average length of contigs decreases while the quality of each contig increases. Three stringency levels can be set:

    - **Low.**

    - **Medium.**

    - **High.**

    The stringency settings Low, Medium and High are based on the following score values (mt=match, ti=transition, tv=transversion, un=unknown):

Score values			
	Low	Medium	High
Match (mt)	2	2	2
Transversion (tv)	-6	-10	-20
Transition (ti)	-2	-6	-16
Unknown (un)	-2	-6	-16
Gap	-8	-16	-36

Score Matrix					
	A	C	G	T	N
A	mt	tv	ti	tv	un
C	tv	mt	tv	ti	un
G	ti	tv	mt	tv	un
T	tv	ti	tv	mt	un
N	un	un	un	un	un

- **Conflicts resolution.** If there is a conflict, i.e. a position where aligned sequences 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 (conflicts are registered already when two nucleotides differ).

  - **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the aligned sequences (nucleotide ambiguity is registered when two nucleotides differ). For an overview of ambiguity codes, see Appendix H.

  - **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 be highlighted for all options. Furthermore, conflicts will be marked with an annotation on each contig sequence which are preserved 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.

- **Trimming options.** When aligning sequences to 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 26.3 for more information about trimming).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**. This will start the assembly process. See section 26.8 on how to use the resulting contigs.

26.7   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 Analysis (⟨⟩)| 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.

Often, the results of the assembly will be better if the sequences are trimmed first (see section 26.3.1).

When the elements are selected, click **Next**, and you will see the dialog shown in figure 26.19
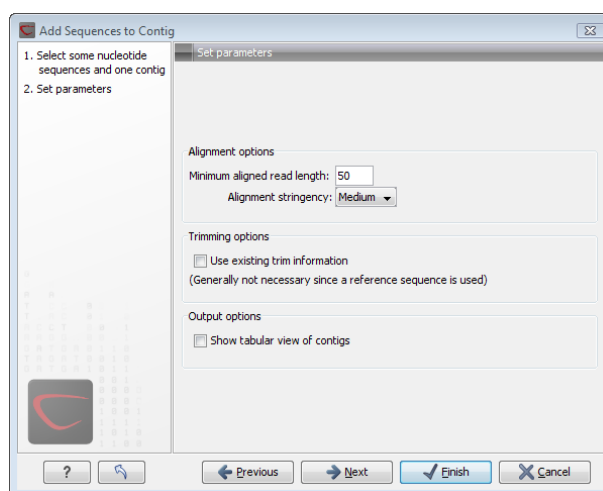
Figure 26.19: *Setting assembly parameters when assembling to an existing contig.*

The options in this dialog are similar to the options that are available when assembling to a reference sequence (see section 26.6).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**. This will start the assembly process. See section 26.8 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.

26.8   View and edit contigs

The result of the assembly process is one or more contigs where the sequence reads have been aligned (see figure 26.20).

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 contig. 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 contig: simply drag the edge of the faded area as shown in figure 26.21.
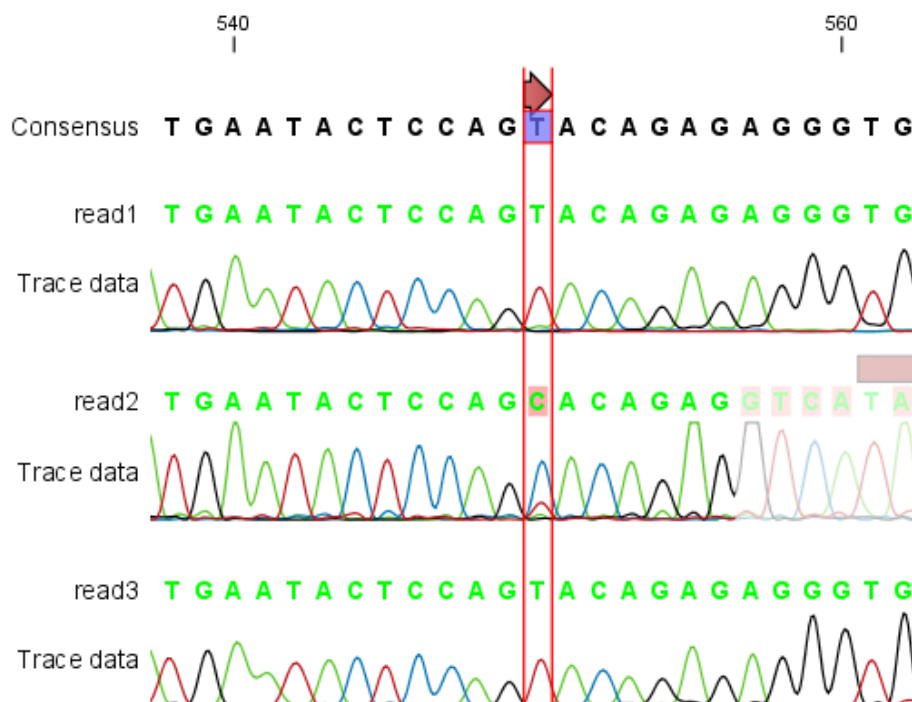
Figure 26.20: *The view of a contig. Notice that you can zoom to a very detailed level in contigs.*
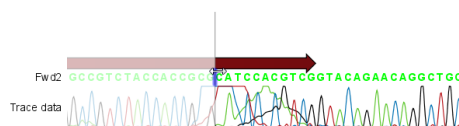
Figure 26.21: *Dragging the edge of the faded area.*

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", "Low" or "Packed". 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 26.8.1

If you find out that the reversed reads should have been the forward reads and vice versa, you can reverse complement the whole contig(imagine flipping the whole contig):

> **right-click in the empty white area of the contig** | **Reverse Complement**

26.8.1 View settings in the Side Panel

Apart from this the view resembles that of alignments (see section 29.2) but has some extra preferences in the **Side Panel**:

Apart from this the view resembles that of alignments but has some extra preferences in the **Side Panel**:

- **Read layout.** This section appears at the top of the **Side Panel** when viewing a stand alone read mapping:

– **Compactness**The compactness setting options let you control the level of detail to be displayed. This setting affects many of the other settings in the **Side Panel** as well as the general behavior of the view. For example: if the compactness is set to **Compact**, you will not be able to see quality scores or annotations on the reads, even if these are turned on via the Nucleotide info section of the Side Panel. You can change the Compactness setting in the Side Panel directly, or you can use the shortcut: press and hold the Alt key while you scroll with the mouse wheel or touchpad.

  * **Not compact.** This allows the mapping to be viewed full detail, including quality scores and trace data for the reads, where this is relevant. To view such information, additional viewing options under the **Nucleotide info** view settings must also selected. For further details on these, please see section 26.1.2 and section 17.1.

  * **Low.** Hides trace data, quality scores 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 cannot be seen.

  * **Compact.** Even less space between the reads.

  * **Packed.** All the other compactness settings will stack the reads on top of each other, but the packed setting will use all space available for displaying the reads. When zoomed in to 100%, you can see the residues but when zoomed out the reads will be represented as lines just as with the Compact setting. The packed mode is very useful when viewing large amounts of data. However certain functionality possible with other views are not available in packed view. For example, no editing of the read mapping or selections of it can be done and color coding changes are not possible. An example of the packed setting is shown in figure 26.22.
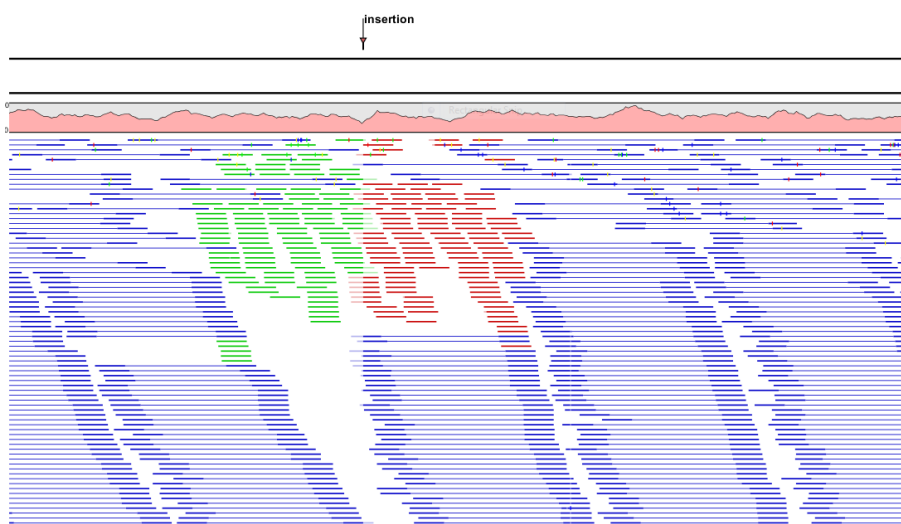
Figure 26.22: *An example of the packed compactness setting.*

– **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. This setting is not relevant when the compactness is packed.

– **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.

– **Show mismatches.** When the compactness is packed, you can highlight mismatches which will get a color according to the Rasmol color scheme. A mismatch is whenever the base is different from the reference sequence at this position. This setting also causes the reads that have mismatches to be floated at the top of the view.

– **Disconnect pairs.** This option will break up the paired reads in the display (they are still marked as pairs - this is just affects the visualization). The reads are marked with colors for the direction (default red and green) instead of the color for pairs (default blue). This is particularly useful when investigating overlapping pairs in packed view and when the strand / read orientation is important.

– **Packed read height.** When the compactness is set to "packed", you can choose the height of the visible reads. When there are more reads than the height specified, an overflow graph will be displayed below the reads. The overflow graph is shown in the same colors as the sequences, and mismatches in reads are shown as narrow horizontal lines in. The colors of the small lines represent the mismatching residue. The color codes for the horizontal lines correspond to the color used for highlighting mismatches in the sequences (red = A, blue = C, yellow = G, and green = T). E.g. a red line with half the height of the blue part of the overflow graph will represent a mismatching "A" in half of the paired reads at this particular position.

– **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.

– **Low coverage threshold.** All regions with coverage up to and including this value are considered low coverage. When clicking the 'Find low coverage' button the next region in the read mapping with low coverage will be selected.

• **Alignment info.** There is one additional parameter:

– **Coverage**: Shows how many sequence reads that are contributing information to a given position in the contig. 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 13.4).

    · **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.

- **Sequence layout.** At the top of the **Side Panel**:

  – **Matching residues as dots** Matching residues will be presented as dots. Only the top sequence will be preserved in its original format.

There are many other viewing options available, both general and aimed as specifice elements of a mapping, which can be adjusted in the View settings. Those covered here were the key ones relevant standard review of mapping results.

26.8.2 Editing the contig

When editing contigs, 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 Main Workbench* all changes are recorded in the history log (see section 14) 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 contig view, you can use **Zoom in** (⊕) to zoom to a greater level of detail than in other views (see figure 26.20). 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 contigs with more than 1000 reads, you can only do single-residue replacements (you can't delete or edit a selection). When the compactness is **Packed**, you cannot edit any of the reads.

26.8.3   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** or set the compactness to **Packed** 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.

26.8.4   Read conflicts

When the contig 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**.

26.8.5   Using the contig

Due to the integrated nature of *CLC Main Workbench* it is easy to use the consensus sequences as input for additional analyses.

You can also right-click the consensus sequence and select **Open Sequence**. This will not create a new sequence but simply let you see the sequence in a sequence view. This means that the

sequence still "belong" to the contig and will be saved together with the contig. It also means that if you add annotations to the sequence, they will be shown in the contig view as well. This can be very convenient e.g. for Primer design ( ).

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**.

26.8.6   Extract parts of a contig

Sometimes it is useful to extract part of a contig for in-depth analysis. This could be the case if you have performed an analysis of a whole genome data set and have found a region that you are particularly interested in analyzing further. Rather than running all further analysis on your full data, you may prefer to run only on a subset of the data. You can extract a subset of your contig data by running the **Extract from Selection** tool on a selected region in your contig. The result of running this tool is a new contig which contains only the reads (and optionally only those that are of a particular type) in your selected region.

To select a region, use the **Selection mode** ( ) (see Section 9.2.3 for a detailed description of the different modes) and select you region of interest in your contig, then right-click. You are now presented with the dialog shown in Figure 26.23.
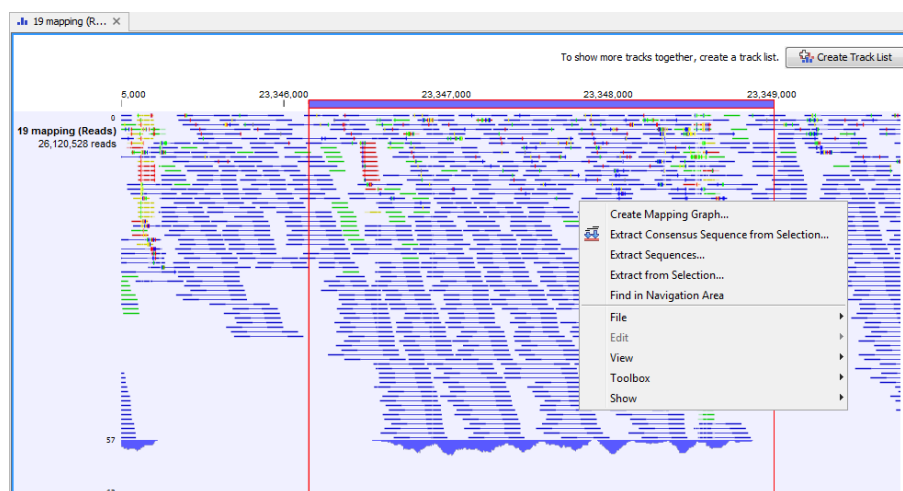
Figure 26.23: *Extracting parts of a contig.*

When you choose the **Extract from Selection** option you are presented by the dialog shown in figure 26.24.

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 status Include intact paired reads When paired reads are placed within the paired dis-
     tance 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,
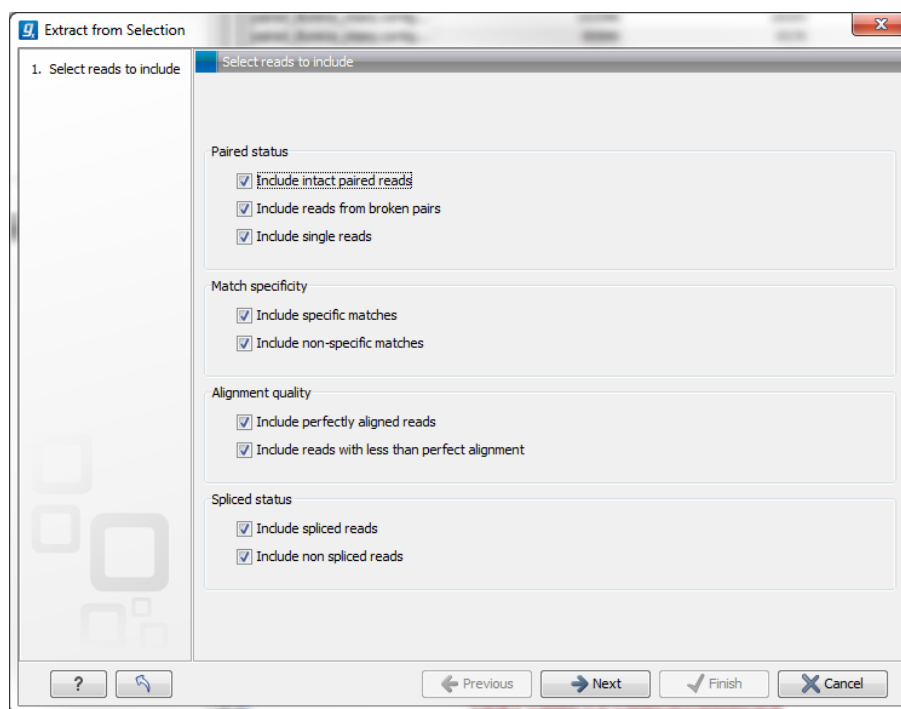
Figure 26.24: *Selecting the reads to include.*

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).

Spliced status Include spliced reads Reads that are across an intron.

Include non spliced reads Reads that are not across an intron.

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 22.1)

You will now have all paired reads from the original mapping in a list.

26.8.7 Variance table

In addition to the standard graphical display of a contig 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 26.25.
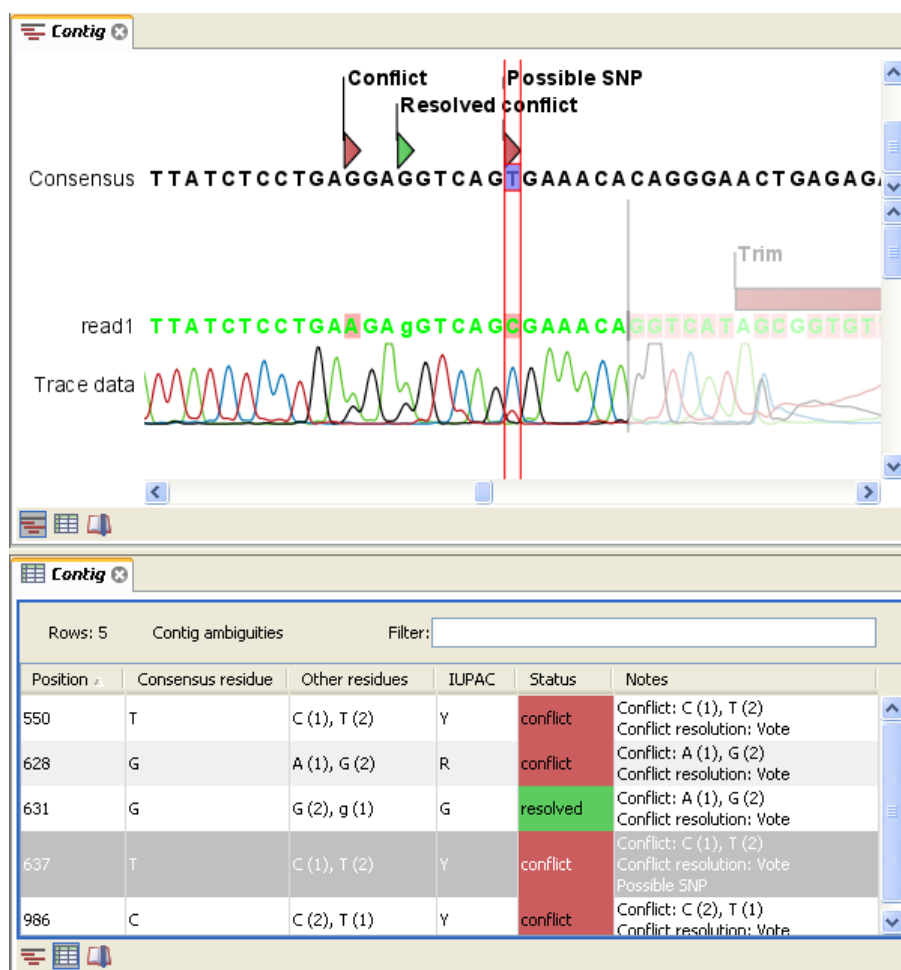

Figure 26.25: *The graphical view is displayed at the top. At the bottom the conflicts are shown in a table. At the conflict at position 637, the user has entered a comment in the table. This comment is now also reflected on the tooltip of the conflict annotation in the graphical view above.*

The table has the following columns:

- **Reference position.** The position of the conflict measured from the starting point of the reference sequence.

- **Consensus position.** The position of the conflict measured from the starting point of the consensus 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 26.25, 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 H.)

- **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 26.25). The comments are saved when you **Save** (⇤).

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 contig, 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.

26.9   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 Analysis** (⚇)| **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 26.26

In this dialog, you can choose:

- **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 26.4. The consensus sequence of the contig will be ignored.
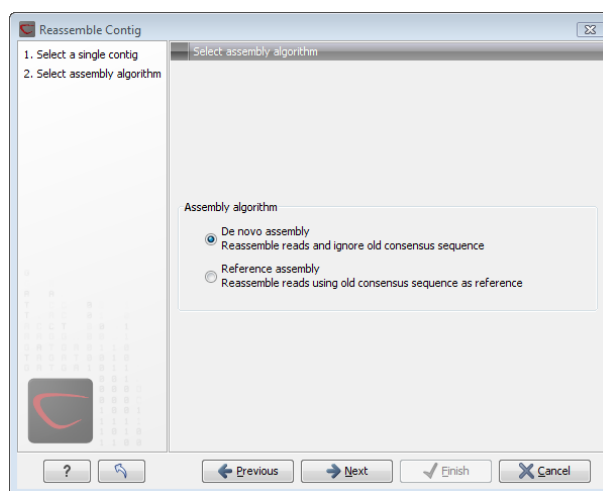
Figure 26.26: *Re-assembling a contig.*

- **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 26.6.

When you click **Finish**, a new contig is created, so you do not lose the information in the old contig.

26.10 Secondary peak calling

CLC Main 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 Main 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 detection investigates any secondary high peaks in the same interval as the already called peaks. The peaks must have a peak shape in order to be considered (i.e. a fading signal from the previous peak will be ignored).

Regions that are trimmed (i.e. covered by trim annotations) are ignored in the analysis (section 26.3).

When a secondary peak is called, the residue is change to an ambiguity character to reflect that two bases are possible at this position, and optionally an annotation is added at this position.

To call secondary peaks:

> **select sequence(s)** | **Toolbox in the Menu Bar** | **Sequencing Data Analysis (🔬)**| **Call Secondary Peaks (△)**

This opens a dialog where you can add the sequences to be analyzed.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 26.27.

The following parameters can be adjusted in the dialog:

- **Fraction of max peak height for calling.** Adjust this value to specify how high the secondary
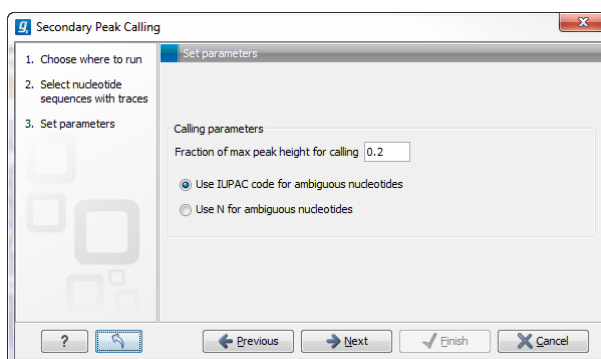
Figure 26.27: *Setting parameters secondary peak calling.*

peak must be to be called.

- **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 H).

Clicking **Next** allows you to add annotations. In addition to changing the actual sequence, annotations can be added for each base that has been called. The annotations hold information about the fraction of the max peak height.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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 27

Expression analysis

Contents

The *CLC Main 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 *CLC Main Workbench* is not able to calculate expression levels based on the raw sequence data. This analysis has to be performed with the *CLC Genomics Workbench*. The result of this analysis can be imported and further analyzed in the *CLC Main Workbench*.

The *CLC Main 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.

27.1   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 Main 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 *CLC Main Workbench* is not able to calculate expression levels based on the raw sequence data. This analysis has to be performed with the *CLC Genomics Workbench*. The result of this analysis can be imported and further analyzed in the *CLC Main Workbench*.

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.

27.1.1   Setting up an experiment

To set up an experiment:

> **Toolbox | Transcriptomics Analysis (** 🖼️ **)| Set Up Experiment (** 🖼️ **)**

Select the samples that you wish to use by double-clicking or selecting and pressing the **Add** (➡️) button (see figure 27.1).

Note that we use "samples" as the general term for both microarray-based sets of expression values and sequencing-based sets of expression values (e.g. an expression track from RNA-Seq).

Clicking **Next** shows the dialog in figure 27.2.

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.
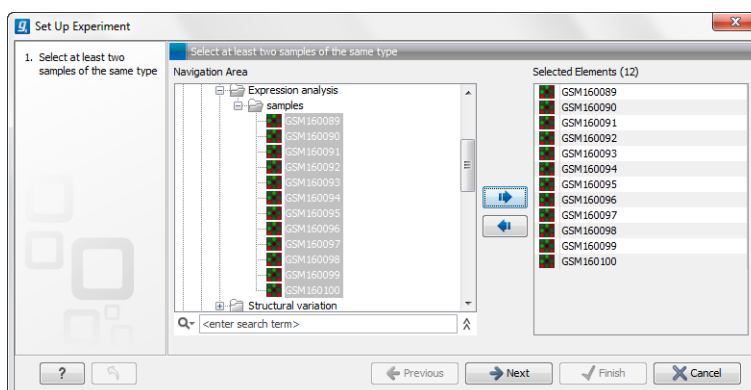
Figure 27.1: *Select the samples to use for setting up the experiment.*
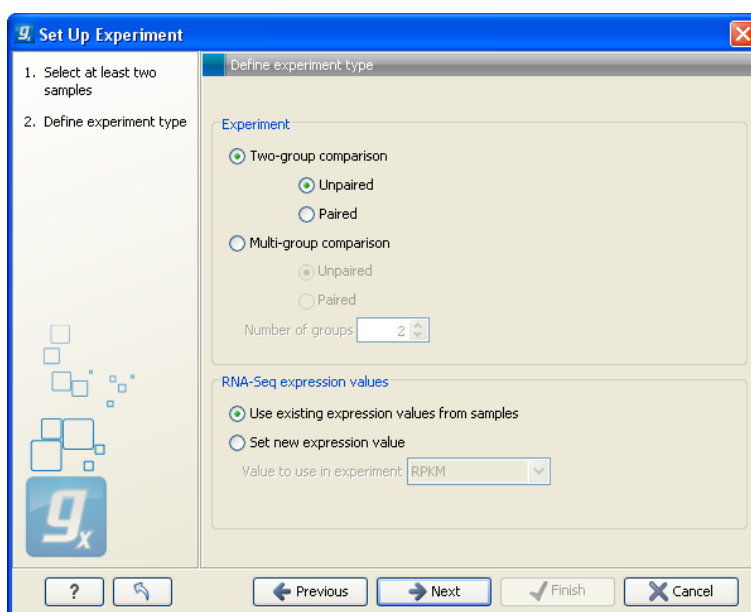
Figure 27.2: *Defining the number of groups.*

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 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 subsequent analyses.

Clicking **Next** shows the dialog in figure 27.3.

Depending on the number of groups selected in figure 27.2, 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** (❌)

Figure 27.3: *Naming the groups.*

button. If you need more groups, simply click **Add New Group**.

Click **Next** when you have named the groups, and you will see figure 27.4.
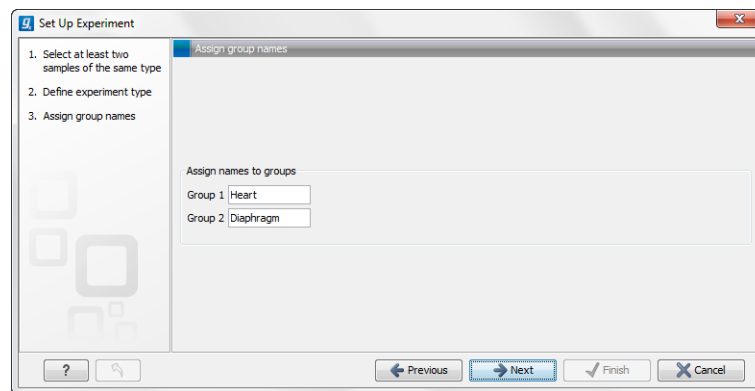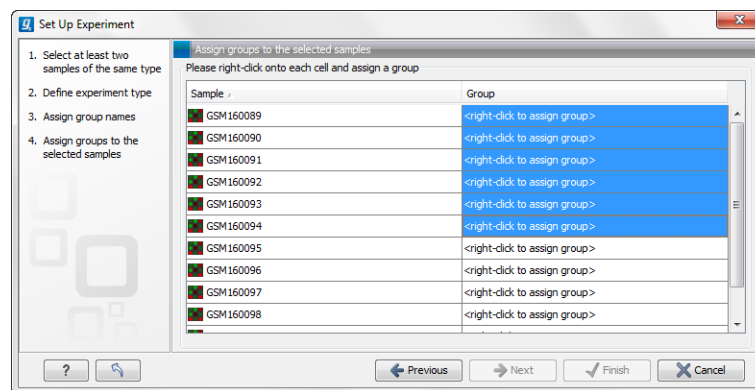

Figure 27.4: *Putting the samples into groups.*

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 27.2, 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 15.2). If not, click **Finish**.

27.1.2 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 27.5).
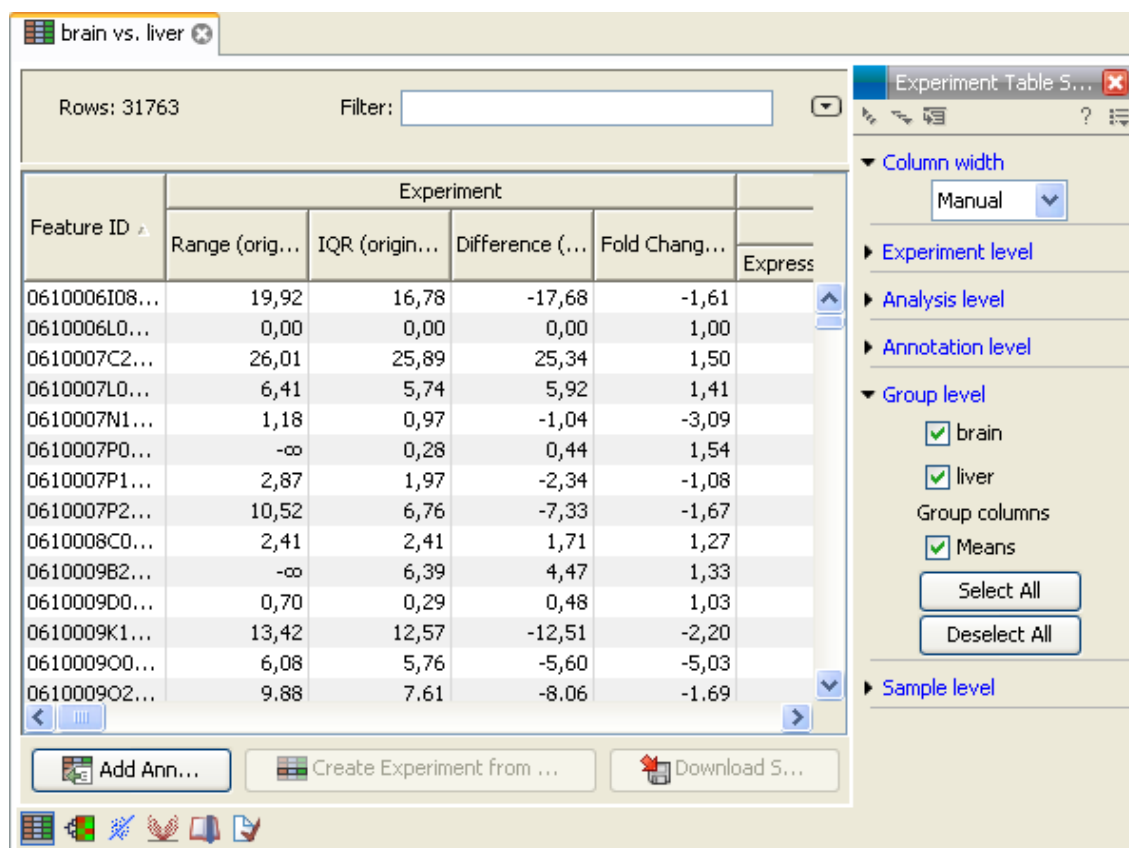
Figure 27.5: *Opening the experiment.*

For a general introduction to table features like sorting and filtering, see section 15.3.

Unlike other tables in *CLC Main 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 11.6).

Whenever you perform analyses like normalization, transformation, statistical analysis etc, new 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 27.6).
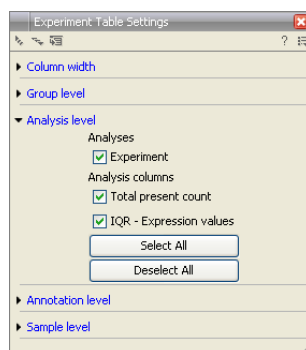

Figure 27.6: *The initial view of the experiment level for a two-group experiment.*

Initially, it has one header for the whole **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 before 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 before 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.

If the samples used are Affymetrix GeneChips samples and have 'Present calls' there will also be a 'Total present count' column containing the number of present calls for all samples.

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 27.2.3 and 27.2.2, 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 27.4.5. 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 27.4), 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 27.4).

An example of a more elaborate analysis level is shown in figure 27.7.

Annotation level

If your experiment is annotated (see section 27.1.3), the annotations will be listed in the **Annotation level** group as shown in figure 27.8.

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 11.6).
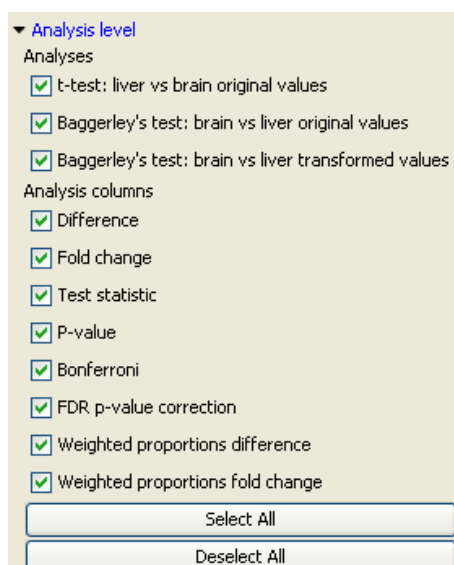
Figure 27.7: *Transformation, normalization and statistical analysis has been performed.*
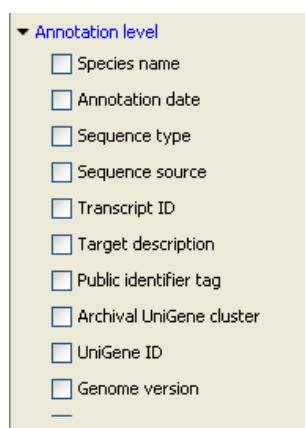

Figure 27.8: *An annotated experiment.*

Group level

At the group level, you can show/hide entire groups (*Heart* and *Diaphragm* in figure 27.5). 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 27.2.3 and 27.2.2, 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 all the columns in the samples.

If you have performed normalization or transformation (see sections 27.2.3 and 27.2.2, respectively), the normalized and transformed values will also appear.
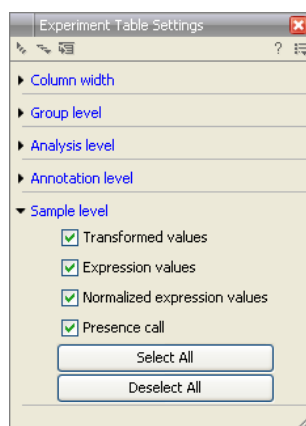
An example is shown in figure 27.9.

Figure 27.9: *Sample level when transformation and normalization has been performed.*

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 15.3).

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 Experiment from Selection** (⊞) button at the bottom of the table (see figure 27.10).

Figure 27.10: *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 27.11).

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 17. You can now use the downloaded sequences for further analysis in the Workbench, e.g. performing BLAST searches and designing primers for QPCR experiments.

Figure 27.11: *Select sequences and press the download button.*

27.1.3 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 J for information about the different annotation file formats that are supported *CLC Main 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 Main Workbench* in section J. In order to associate an annotation file with an experiment, either select the annotation file when you set up the experiment (see section 27.1.1), or click:

Toolbox | Transcriptomics 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 27.1.2. You can also add annotations by pressing the **Add Annotations** ( ) button at the bottom of the table (see figure 27.12).

Figure 27.12: *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 27.13).

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 (experiment or sample) as well as for the annotations. Usually the default is right, but for some annotation files, you need to select 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.
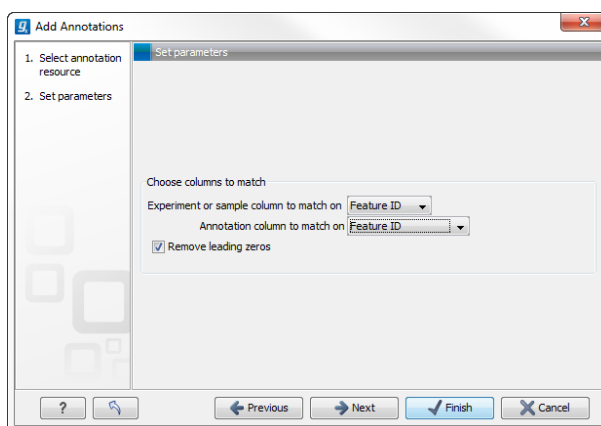
Figure 27.13: *Choosing how to match annotations with samples.*

27.1.4 Scatter plot view of an experiment

At the bottom of the experiment table, you can switch between different views of the experiment (see figure 27.14).

Figure 27.14: *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 27.15.
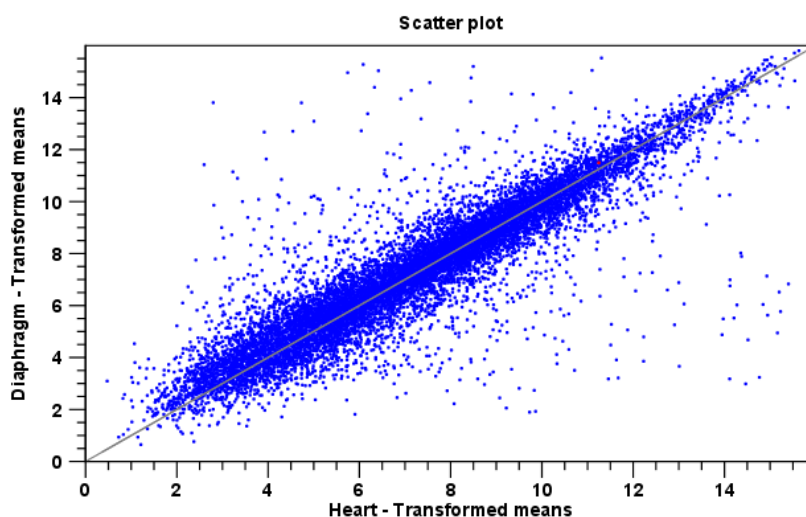
Figure 27.15: *A scatter plot of group means for two groups (transformed expression values).*

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 11.6).

27.1.5 Cross-view selections

There are a number of different ways of looking at an experiment as shown in figure 27.16).

Figure 27.16: *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 27.16, 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 27.17.

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 27.1.2) you typically want to choose 'Difference'.
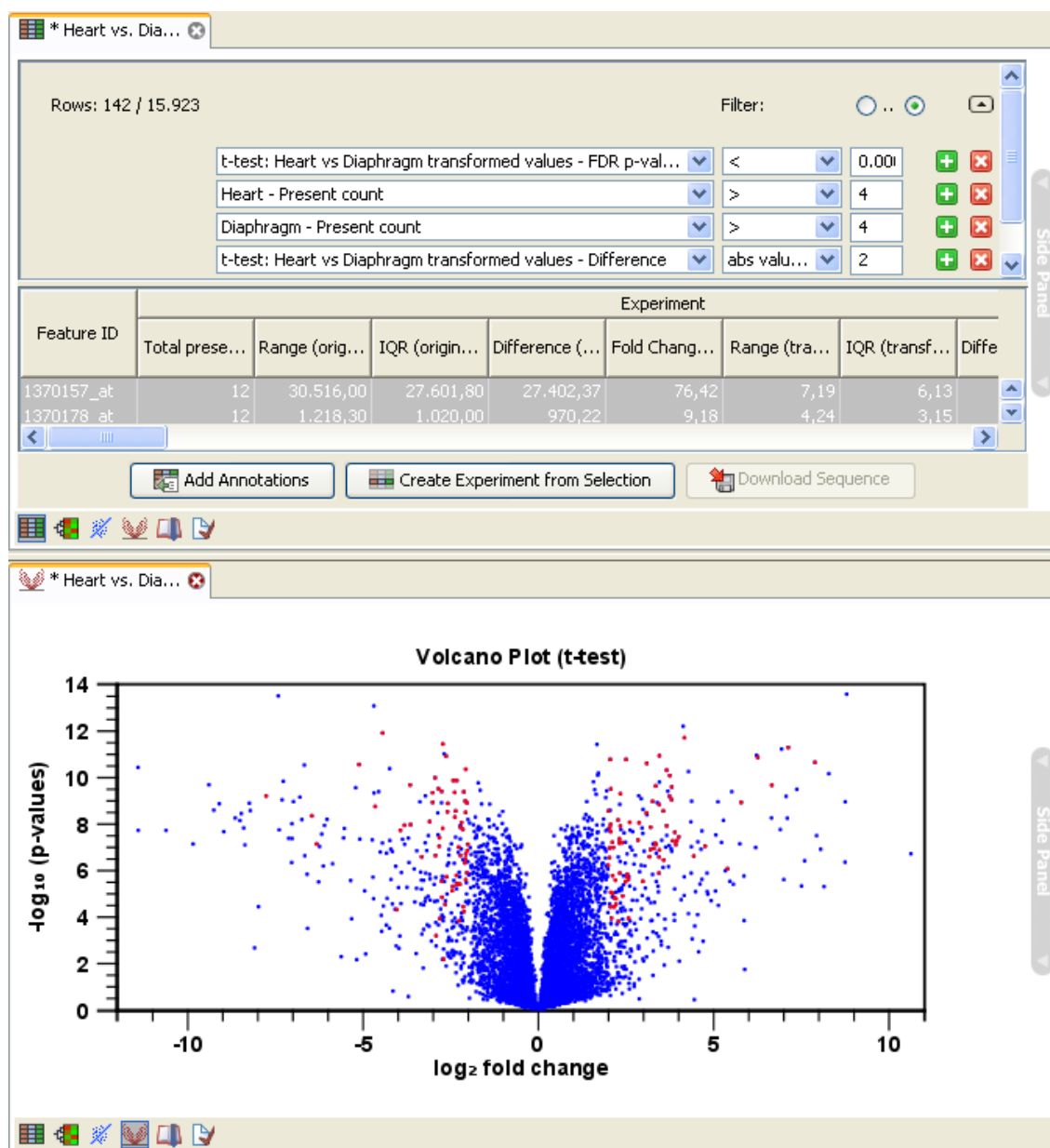
Figure 27.17: *A split view showing an experiment table at the top and a volcano plot at the bottom (note that you need to perform statistical analysis to show a volcano plot, see section 27.4).*

27.2 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).

27.2.1 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 27.18.
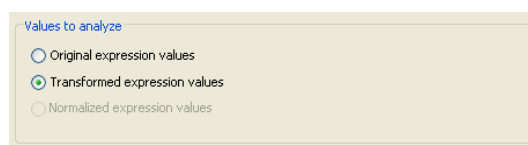

Figure 27.18: *Selecting which version of the expression values to analyze. In this case, the values have not been normalized, so it is not possible to select normalized values.*

In this case, the values have not been normalized, so it is not possible to select normalized values.

27.2.2 Transformation

The *CLC Main Workbench* lets you transform expression values based on logarithm and adding a constant:

> **Toolbox | Transcriptomics Analysis (▦)| Transformation and Normalization | Transform (▦)**

Select a number of samples ((▦) or (▦)) or an experiment (▦) and click **Next**.

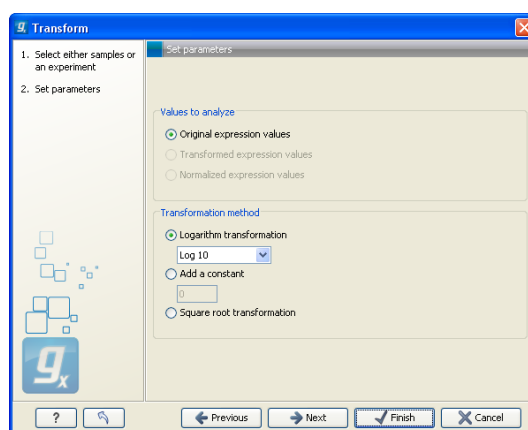This will display a dialog as shown in figure 27.19.

Figure 27.19: *Transforming expression values.*

At the top, you can select which values to transform (see section 27.2.1).

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 15.2). If not, click **Finish**.

27.2.3 Normalization

The *CLC Main Workbench* lets you normalize expression values.

To start the normalization:

> **Toolbox | Transcriptomics 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 27.20.
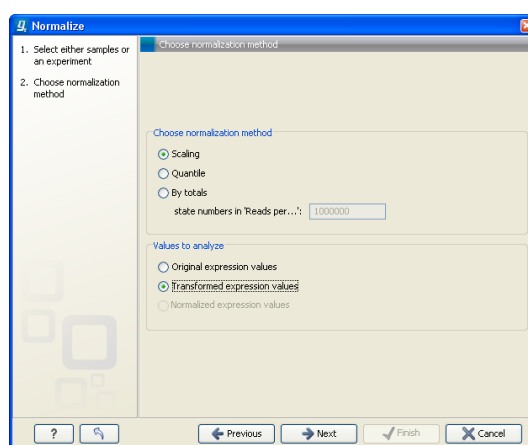
Figure 27.20: *Choosing normalization method.*

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 27.21 and 27.22 show the effect on the distribution of expression values when using scaling or quantile normalization, respectively.
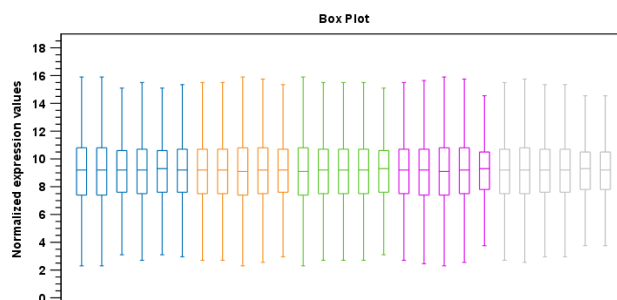

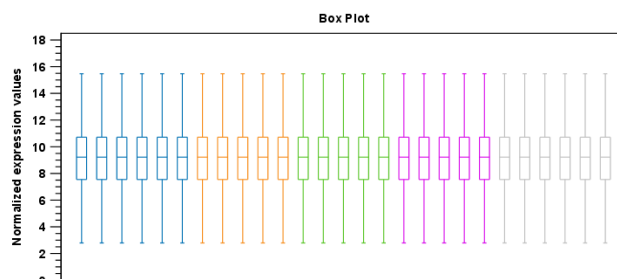Figure 27.21: *Box plot after scaling normalization.*

Figure 27.22: *Box plot after quantile normalization.*

At the bottom of the dialog in figure 27.20, you can select which values to normalize (see section 27.2.1).

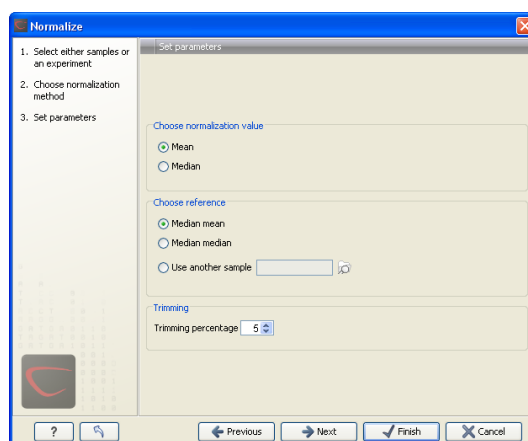Clicking **Next** will display a dialog as shown in figure 27.23.

Figure 27.23: *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 15.2). If not, click **Finish**.

27.3 Quality control

The *CLC Main 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.

27.3.1 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 | Transcriptomics 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 27.24.

Here you select which values to use in the box plot (see section 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.
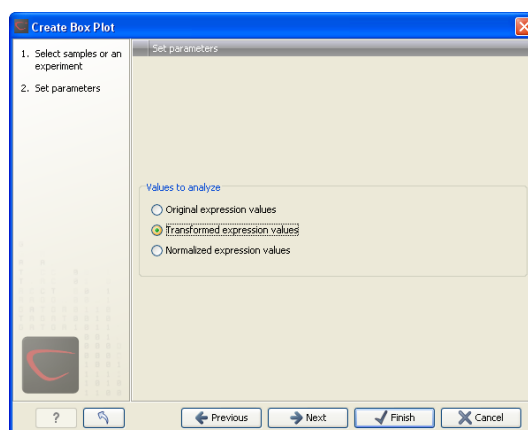
Figure 27.24: *Choosing values to analyze for the box plot.*

Viewing box plots

An example of a box plot of a two-group experiment with 12 samples is shown in figure 27.25.
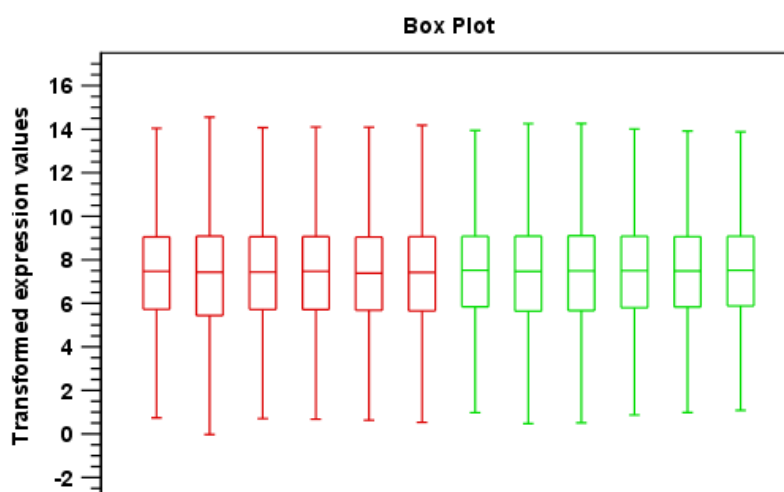

Figure 27.25: *A box plot of 12 samples in a two-group experiment, colored by group.*

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 names are not shown in figure 27.25).

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 27.26).

- **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.

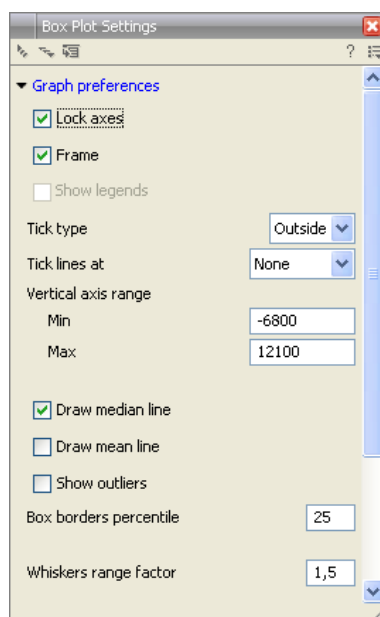- **Show legends.** Shows the data legends.

Figure 27.26: *Graph preferences for a box plot.*

- **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 27.27).

- **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
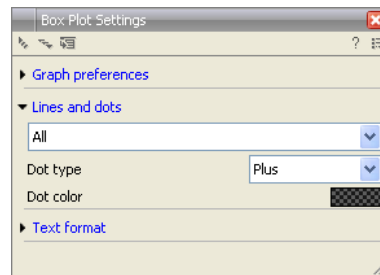
Figure 27.27: *Lines and dot preferences for a box plot.*

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 11.6).

Interpreting the box plot

This section will show how to interpret a box plot through a few examples.

First, if you look at figure 27.28, you can see a 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 27.29 shows a box plot for the same experiment after quantile normalization: the distributions have been brought into par.

In figure 27.30 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.
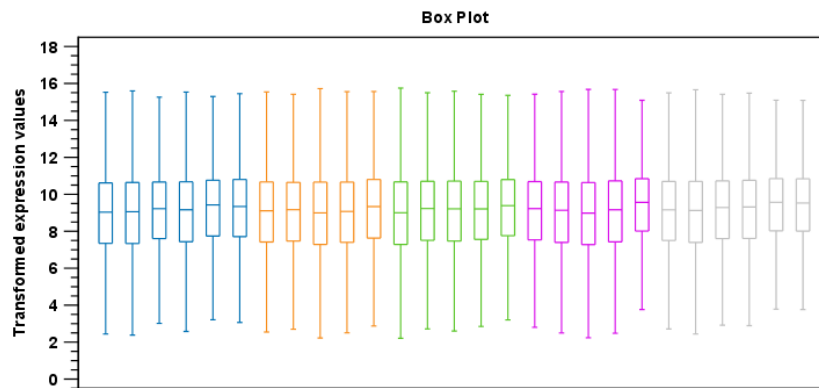
Figure 27.28: *Box plot for an experiment with 5 groups and 27 samples.*
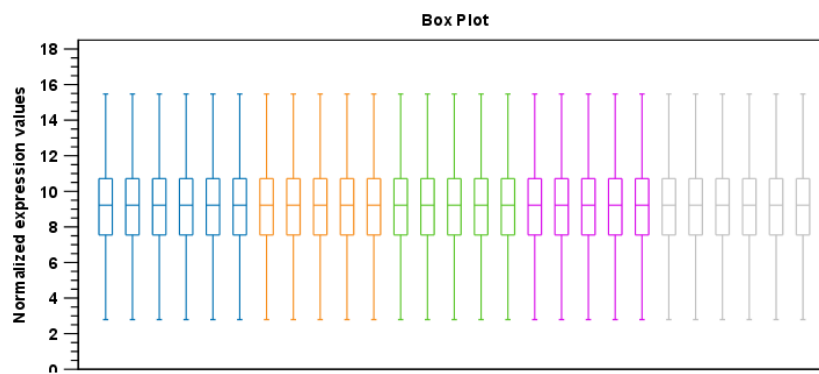
Figure 27.29: *Box plot after quantile normalization.*
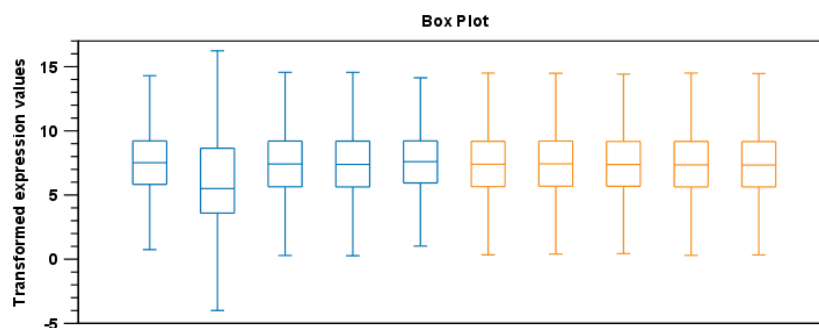
Figure 27.30: *Box plot for a two-group experiment with 5 samples.*

27.3.2   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

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 | Transcriptomics 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 27.31. 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.
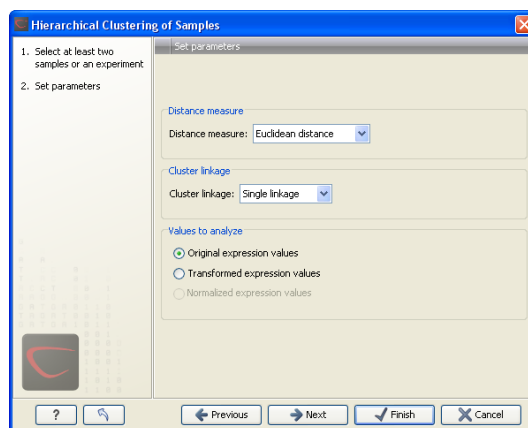

Figure 27.31: *Parameters for hierarchical clustering of samples.*

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

$$|u - v| = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}.$$

- **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 $1 - |Pearson correlation|$ 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 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Result of hierarchical clustering of samples

The result of a sample clustering is shown in figure 27.32.

Figure 27.32: *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 27.33).

Figure 27.33: *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 27.32, 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 27.5.1).

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 27.34).
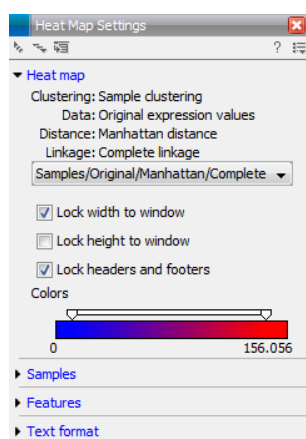

Figure 27.34: *Side Panel of heat map.*

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 27.48).

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
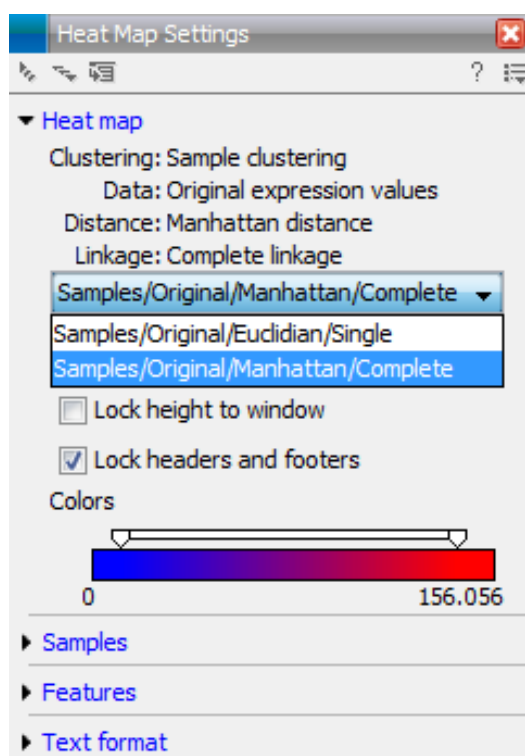
Figure 27.35: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

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, legend, and tree above or below the heatmap. 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 11.6).

27.3.3   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** | **Transcriptomics 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 27.36.
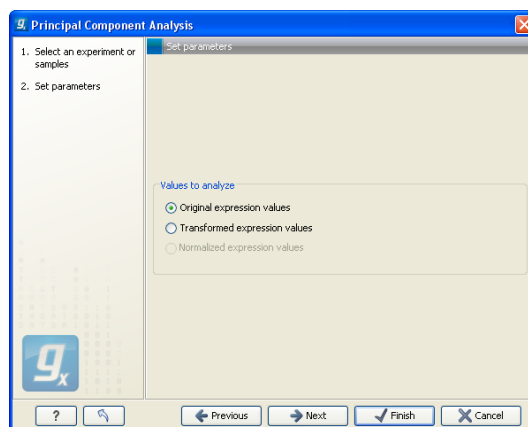
Figure 27.36: *Selcting which values the principal component analysis should be based on.*

In this dialog, you select the values to be used for the principal component analysis (see section 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Principal component analysis plot

This will create a principal component plot as shown in figure 27.37.

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 27.37 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
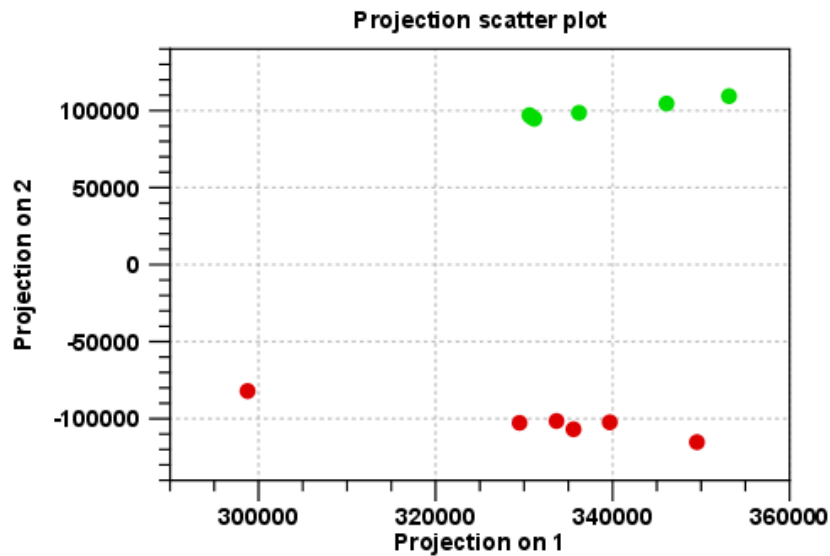
Figure 27.37: *A principal component analysis colored by group.*

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 11.6).

Scree plot

Besides the view shown in figure 27.37, 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 11.6).

27.4   Statistical analysis - identifying differential expression

The *CLC Main 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 types of tests: tests that assume that data consists of counts and compare these or their proportions (described in section 27.4.1 and section 27.4.2) and tests that assume that the data is real-valued, has Gaussian distributions and compare means (described in section 27.4.3). To run the statistical analysis:

> **Toolbox | Transcriptomics Analysis ( )| Statistical Analysis | Empirical Analysis of DGE ( )**

> **Toolbox | Transcriptomics Analysis ( )| Statistical Analysis | On Proportions ( )**

> or   **Toolbox | Transcriptomics Analysis ( )| Statistical Analysis | On Gaussian Data ( )**

For all kinds of statistical analysis you first select the experiment ( ) that you wish to use and click **Next** (learn more about setting up experiments in section 27.1.1).

The first part of the explanation of how to proceed and perform the statistical analysis is divided into three, depending on whether you are doing Empirical analysis of DGE, tests on proportions or Gaussian-based tests. The last part has an explanation of the options regarding corrected p-values which applies to all tests.

27.4.1   Empirical analysis of DGE

The Empirical analysis of DGE tool implements the 'Exact Test' for two-group comparisons developed by Robinson and Smyth [Robinson and Smyth, 2008] and incorporated in the EdgeR Bioconductor package [Robinson et al., 2010]. The test is applicable to count data only, and is

designed specifically to deal with situations in which *many* features are studied simultaneously (e.g. genes in a genome) but where only a *few* biological replicates are available for each of the experimental groups studied. This is typically the case for RNA-seq expression analysis. The test is based on the assumption that the count data follows a Negative Binomial distribution, which in contrast to the Poisson distribution has the characteristic that it allows for a non-constant mean-variance relationship.

The 'Exact Test' of Robinson and Smyth is similar to Fisher's Exact Test, but also accounts for overdispersion caused by biological variability. Whereas Fisher's Exact Test compares the counts in one sample against those of another, the 'Exact Test' compares the counts in one s*et* of count samples against those in another s*et* of count samples. This is achieved by replacing the Hypergeometric distributions of Fisher's Exact Test by Negative binomial distributions, whereby the variability within each of the two groups of samples compared is taken into account. This only works if the dispersions in the two groups compared are identical. As this cannot generally by assumed to be the case for the *original* (nor for the normalized) data, pseudodata for which the dispersion *is* identical is generated from the original data, and the test is carried out on this pseudodata. The generation of the pseudodata is performed simultaneously with the estimation of the dispersion, in an iterative procedure called quantile-adjusted conditional maximum likelihood. Either a single common dispersion for all features may be assumed (as in [Robinson and Smyth, 2008]), or it may be assumed that the dispersion for each feature (e.g. gene) is a 'weighted average' of the common dispersion and feature (e.g. gene) specific dispersions (as suggested in [Robinson and Smyth, 2007]). The weight given to each of the components depends on the number of samples in the groups: the more samples there are in the groups, the higher the weight will be given to the gene-specific component.

The Exact Test in the EdgeR Bioconductor package provides the user with the option to set a large number of parameters. The implementation of the 'Empirical analysis of DGE' algorithm in the Genomics Workbench uses for the most parts the default settings in the edgeR package, version 3.4.0. A detailed outline of the parameter settings is given in section 27.4.1).

Empirical analysis of DGE - implementation parameters

The 'Empirical analysis of DGE' algorithm in the *CLC Main Workbench* is a re-implementation of the "Exact Test", available as part of the EdgeR Bioconductor package.

The parameter values used in the *CLC Main Workbench* implementation are the default values for the equivalent parameters in the EdgeR Bioconductor implementation in all but one case. The exception is the estimateCommonDisp parameter.

The parameter values used in the *CLC Main Workbench* implementation, with reference to the EdgeR function names for clarity, are provided in the table below.

Function in BioC package	Parameter name	Value used and comments
calcNormFactors	method	"TMM"
	refColumn	NULL (automatically selected)
	logratioTrim	0.3
	sumTrim	0.05
	doWeighting	TRUE
	Acutoff	-1e10
estimateCommonDisp	tol	1e-14 (default in edgeR: 1e-6)
	rowsum.filter	Set by user in wizard ("Total count filter cutoff", default 5)
estimateTagewiseDisp	prior.df	10
	trend	"movingave"
	span	NULL
	method	"grid"
	grid.length	11
	grid.range	c(-6, 6)
mglmOneGroup	maxit	50
	tol	1e-10
aveLogCPM	prior.count	2
	dispersion	0.05
exactTest	pair	Set by user in wizard ("Exact test comparisons")
	dispersion	"auto" (tagwise if available, otherwise common)
	rejection.region	"doubletail"
	big.count	900
	prior.count	0.125

Running the Empirical analysis of DGE

The Empirical Analysis of DGE tool should *always be run on the original* counts. This is because the algorithm assumes that the counts on which it operates are Negative Binomially distributed and implicitly normalizes and transforms these counts. If the counts have in any way been altered prior to submitting them to the Empirical Analysis of DGE tool, this assumption is likely to be compromised.

When running the Empirical analysis of DGE tool in the Genomics workbench, the user is asked to specify two parameters related to the estimation of the dispersion (Figure 27.38). Of these, the 'Total count filter cut-off' specifies which features should be considered when estimating the common dispersion component. Features for which the counts across all samples is low are likely to contribute mostly with noise to the estimation, and features with a lower cummulative count across samples than the value specified will be ignored. When the check-box 'Estimate tag-wise dispersions' is checked, the dispersion estimate for each gene will be a weighted combination of the tag-wise and common dispersion, if the check-box is un-ticked the common dispersion will be used for all genes.

The Empirical analysis of DGE may be carried out between all pairs of groups (by clicking the 'All pairs' button) or for each group against a specified reference group (by clicking the 'Against reference' button), Figure 27.39. 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). The user can specify if Bonferroni and FDR corrected p-values should be calculated
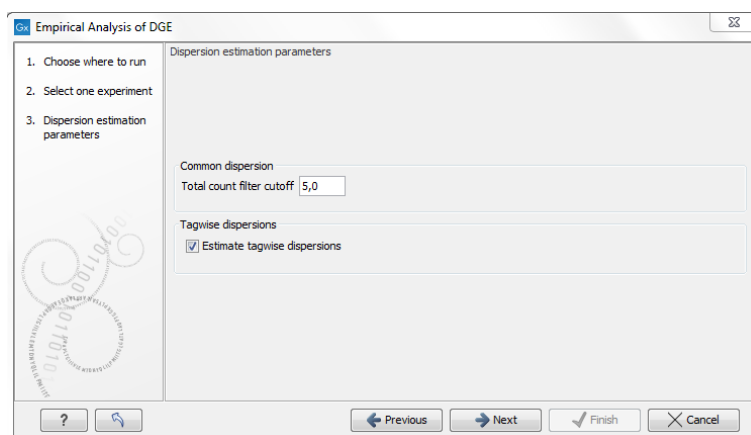
Figure 27.38: *Empirical analysis of DGE: setting the parameters related to dispersion.*
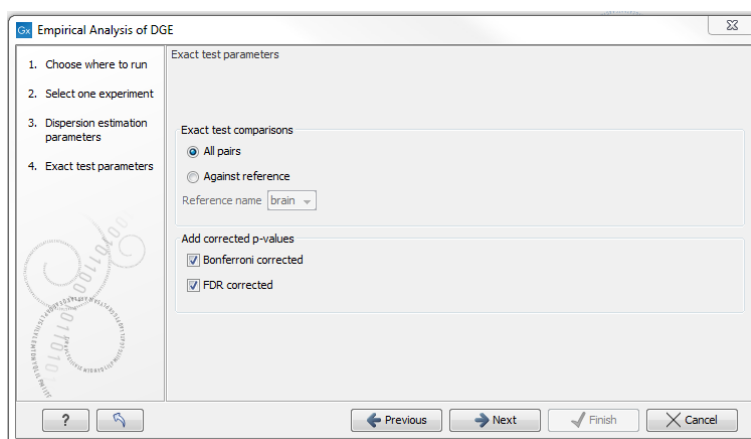
(see Section 27.4.4).

Figure 27.39: *Empirical analysis of DGE: setting comparisons and corrected p-value options.*

When the Empirical analysis of DGE is run three columns will be added to the experiment table for each pair of groups that are analyzed: the 'P-value', 'Fold change' and 'Weighted difference' columns. The 'P-value' holds the p-value for the Exact test. The 'Fold Change' and 'Weighted difference' columns are both calculated from the estimated 'average cpm (counts per million)' values of each of the groups. The estimated 'average cpm' values are values that are derived internally in the Exact Test algorithm. They depend on both the sizes of the samples, the magnitude of the counts and on the estimated negative binomial dispersion, so they cannot be obtained from the original counts by simple algebraic calculations. The 'Fold Change' will tell you how many times bigger the average cpm value of group 2 is relative to that of group 1. If the the average cpm value of group 2 is bigger than that of group 1 the fold change is the average cpm value of group 2 divided by that of group 1. If the average cpm value of group 2 is smaller than that of group 1 the fold change is the average cpm value of group 1 divided by that of group 2 with a negative sign. The 'weighted difference' column contains the difference between the average cpm value of group 2 and the average cpm value of group 1. In addition to the three automatically added columns, columns containing the Bonferroni and FDR corrected p-values will be added if that was specified by the user.

27.4.2 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 27.1.1), 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 sp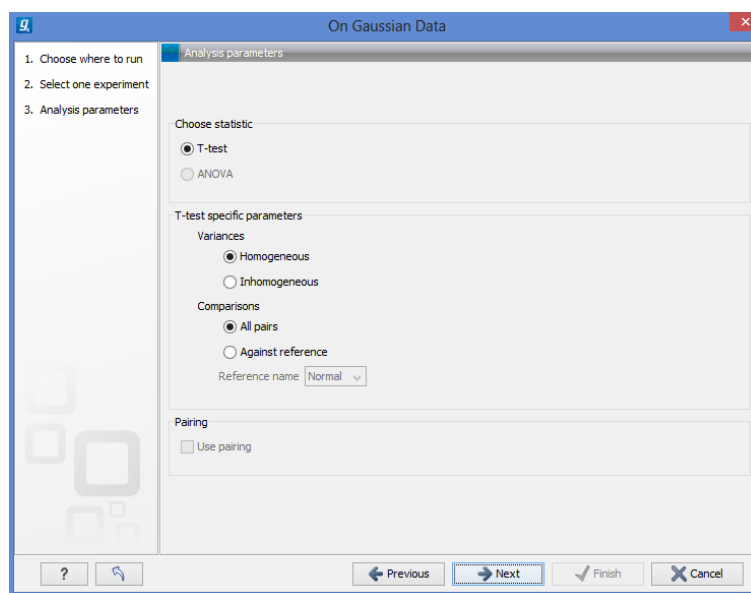ecify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

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 27.4.4).

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 'Weighted proportions 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 27.4.4).

27.4.3 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 27.40.

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

Figure 27.40: *Selecting a t-test.*

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 27.1.1) 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 27.4.4).

ANOVA

For experiments with more than two groups you can choose **T-test** as described above, or **ANOVA** as shown in figure 27.41.

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 27.1.1) 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.
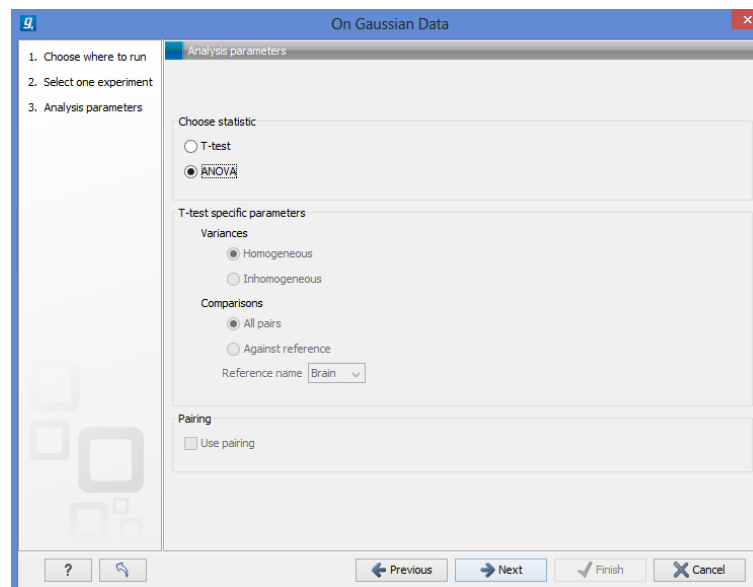
Figure 27.41: *Selecting ANOVA.*

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 before 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 before 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 27.4.4).

27.4.4 Corrected p-values

Clicking **Next** will display a dialog as shown in figure 27.42.

At the top, you can select which values to analyze (see section 27.2.1).

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
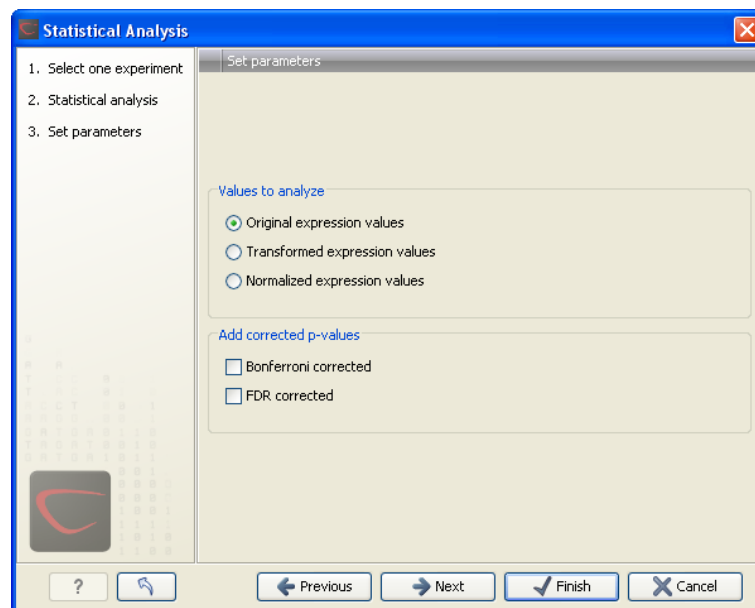
Figure 27.42: *Additional settings for the statistical analysis.*

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 Main Workbench* is that of [Benjamini and Hochberg, 1995].

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Note that if you have already performed statistical analysis on the same values, the existing one will be overwritten.

27.4.5 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 27.1.2). 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 15.3).

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 27.1.4, 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.

An example of a volcano plot is shown in figure 27.43.
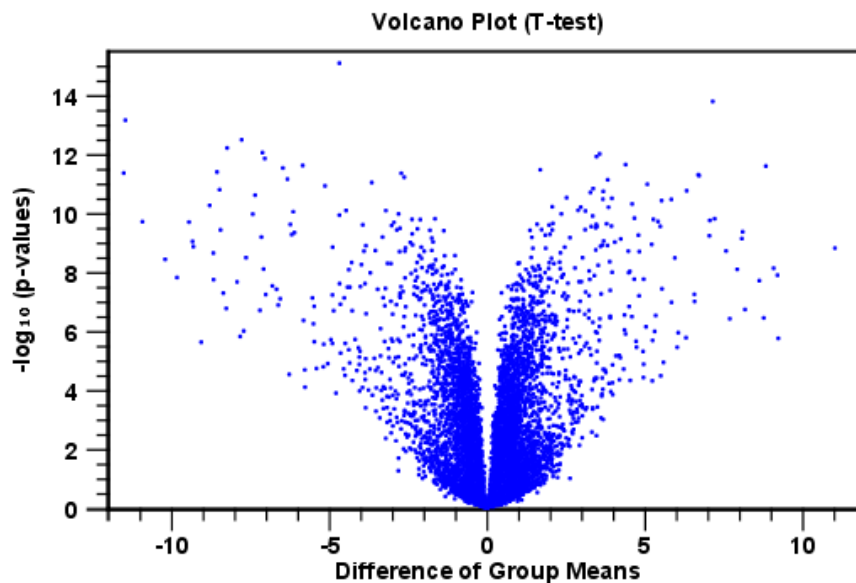
Figure 27.43: *Volcano plot.*

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 27.1.2).

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 9.2).

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 27.1.2.

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 11.6).

27.5   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.

27.5.1   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** | **Transcriptomics Analysis (**![icon]**)**| **Feature Clustering** | **Hierarchical Clustering of Features (**![icon]**)**

Select at least two samples ((![icon]) or (![icon])) or an experiment (![icon]).

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 27.1.2.

Clicking **Next** will display a dialog as shown in figure 27.44. 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**:
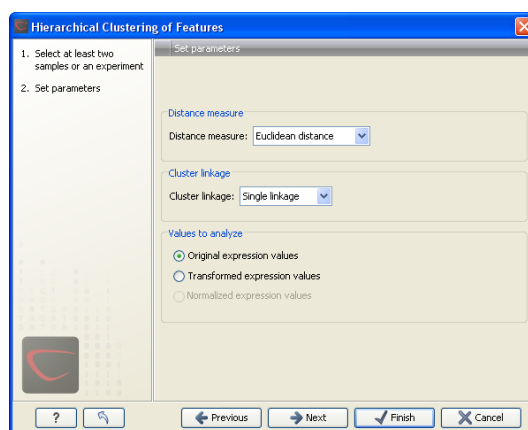
Figure 27.44: *Parameters for hierarchical clustering of features.*

- **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 $1 - |Pearson correlation|$ 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 27.2.1). Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Result of hierarchical clustering of features

The result of a feature clustering is shown in figure 27.45.
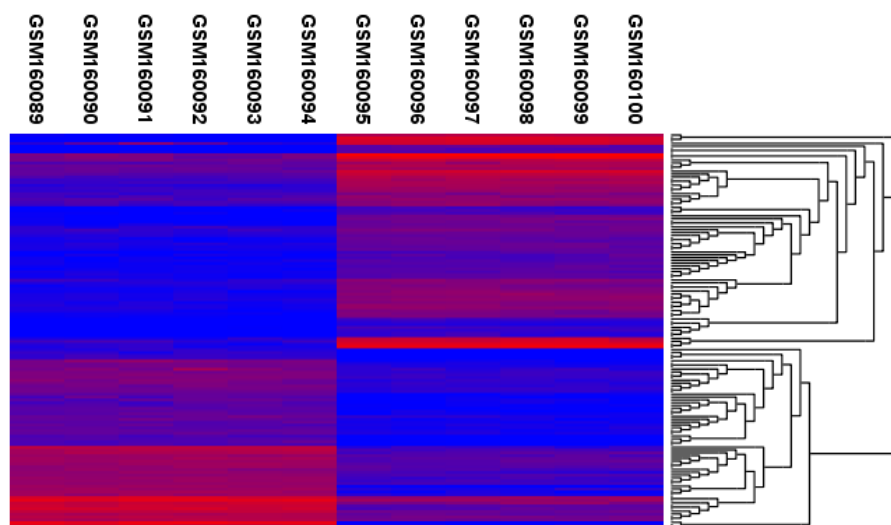
Figure 27.45: *Hierarchical clustering of features.*

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 27.46).

Figure 27.46: *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 27.45). 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 27.47).
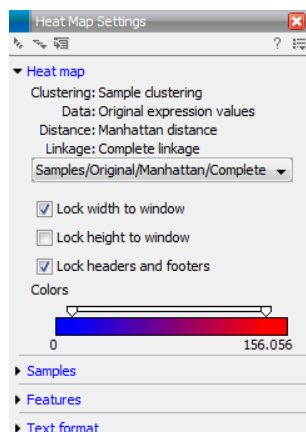
Figure 27.47: *Side Panel of heat map.*

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 27.48).
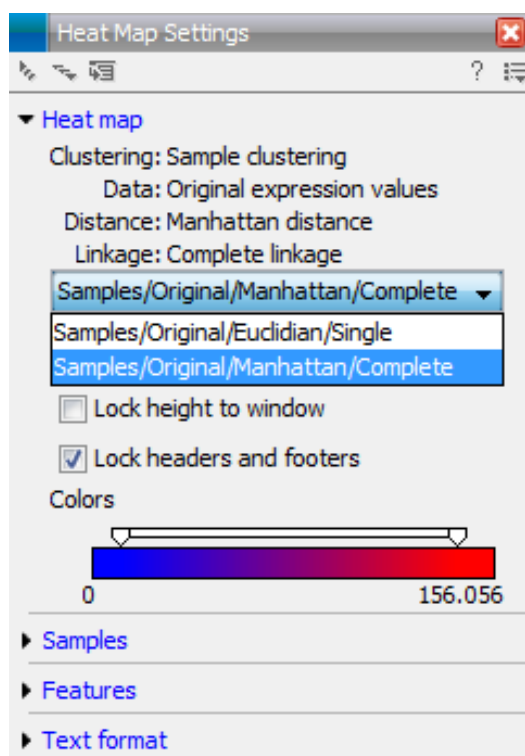
Figure 27.48: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

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, legend, and tree above or below the heatmap. 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 11.6).

27.5.2 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** | **Transcriptomics 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 27.1.2.

Clicking **Next** will display a dialog as shown in figure 27.49.

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 $Z = (z_1, z_2, z_3)$,
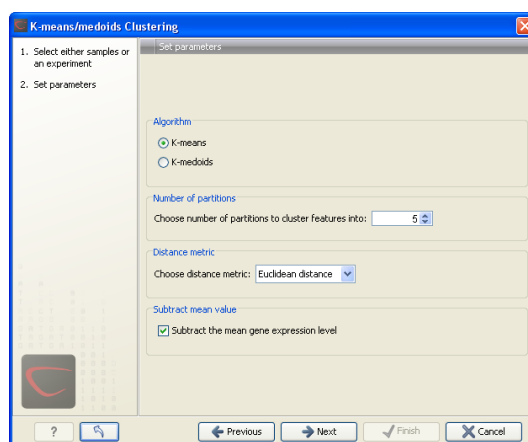
Figure 27.49: *Parameters for k-means/medoids clustering.*

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 27.50.
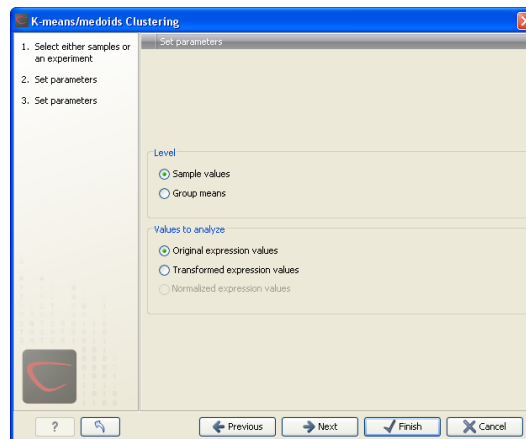
Figure 27.50: *Parameters for k-means/medoids clustering.*

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 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

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 27.49) - there is one graph per cluster. Using drag and drop as explained in section 9.1.6, you can arrange the views to see more than one graph at the time.

Figure 27.51 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 27.1.5.
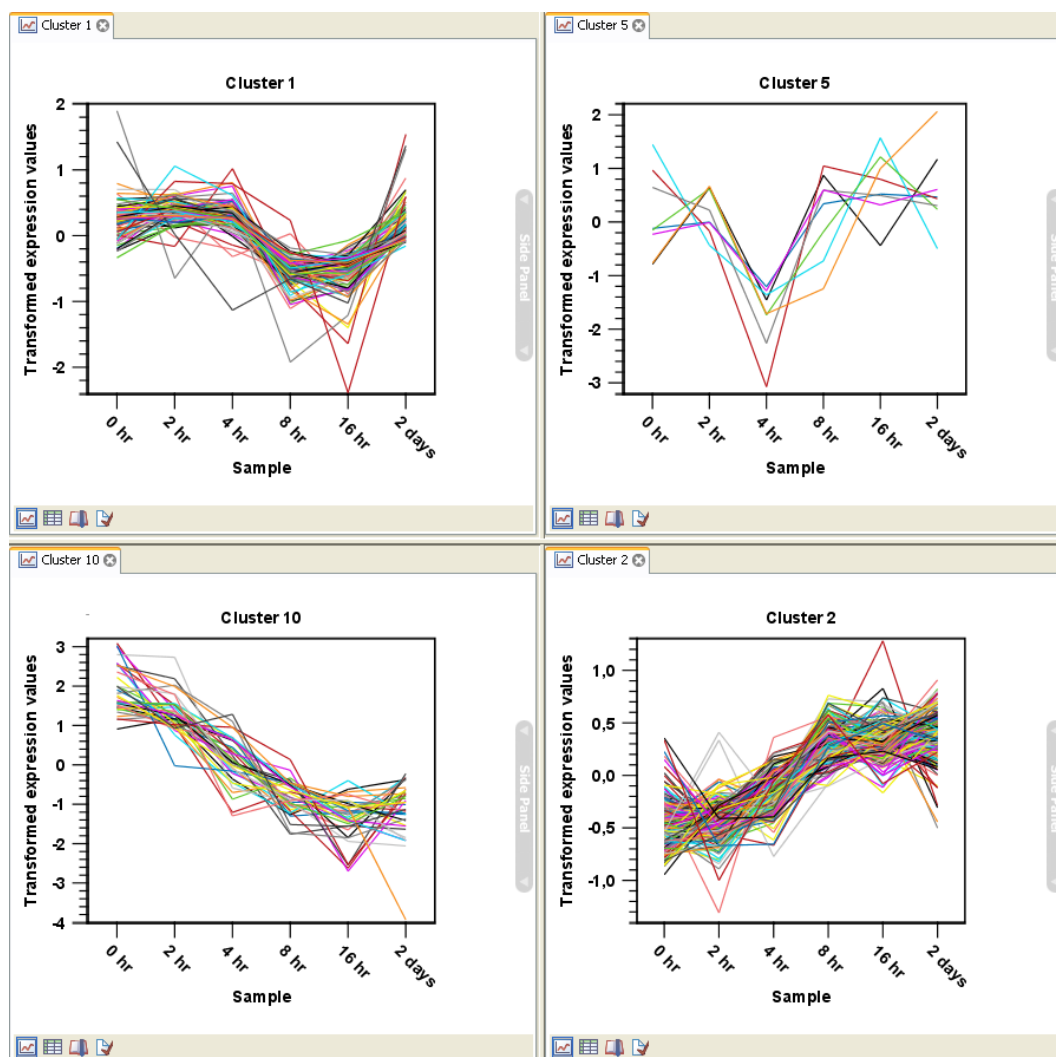
Figure 27.51: *Four clusters created by k-means/medoids clustering.*

27.6 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 27.1.3.

27.6.1 Hypergeometric tests on annotations

The first approach to using annotations to extract biological information is the hypergeometric annotation test. This test measures the extent 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 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 choosing to keep only those features with FDR corrected p-values <0.05 and a fold change 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 | Transcriptomics 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 27.1.2).

Click **Next**. This will display the dialog shown in figure 27.52.
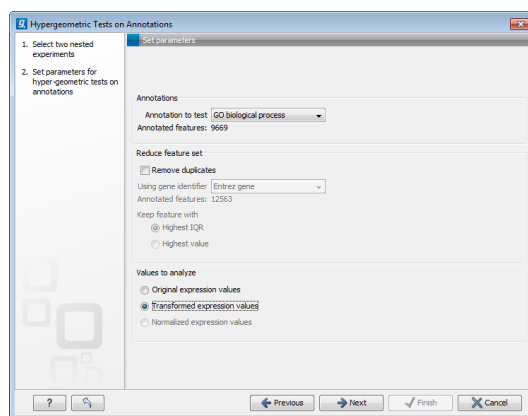
Figure 27.52: *Parameters for performing a hypergeometric test on annotations*

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.

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 the basis on which the feature set will be reduced:

- **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 27.2.1). Only features that have a numerical value assigned to them will be used for the analysis. That is, any feature which

has a value of plus infinity, minus infinity or NaN will not be included in the feature list taken into the test. Thus, the choice of value at this step can affect the features that are taken forward into the test in two ways:
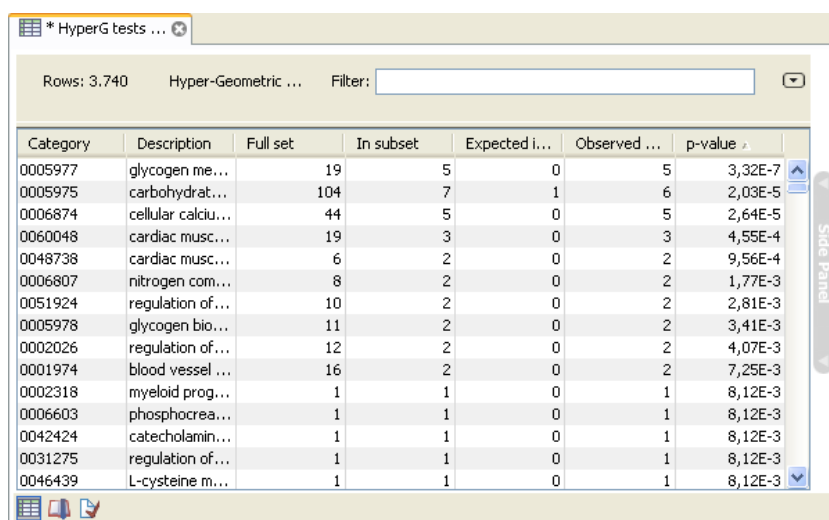
- If there are features with values of plus infinity, minus infinity or NaN, those features will not be taken forward into the test. This can be a consideration when choosing transformed values, where the mathematical manipulations involved may lead to such values.

- If you chose to remove duplicates, then the value type you choose here is the value used for checking the highest IQR or value to determine which feature is taken forward into the test.

The final number of features used for the test is reported in this history view of the test results.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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 27.53.

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 27.53: *The result of testing 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.

- **Full set**. The number of features in the original experiment (not the subset) with this category. (Note that this is after removal of duplicates).

- **In subset**. The number of features in 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.

27.6.2 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 | Transcriptomics Analysis ( )| Annotation Test | Gene Set Enrichment Analysis (GSEA) ( )

Select an experiment and click **Next**.

Click **Next**. This will display the dialog shown in figure 27.54.
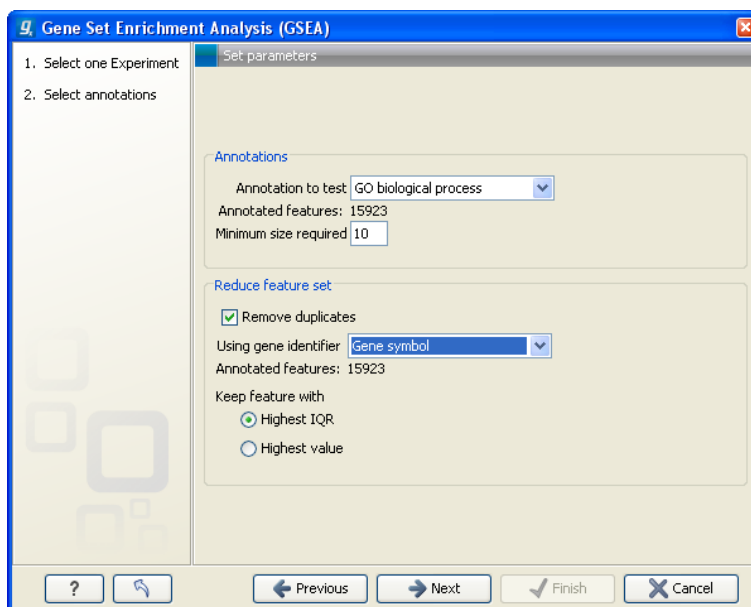
Figure 27.54: *Gene set enrichment analysis on GO biological process*

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.

– **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 27.55.
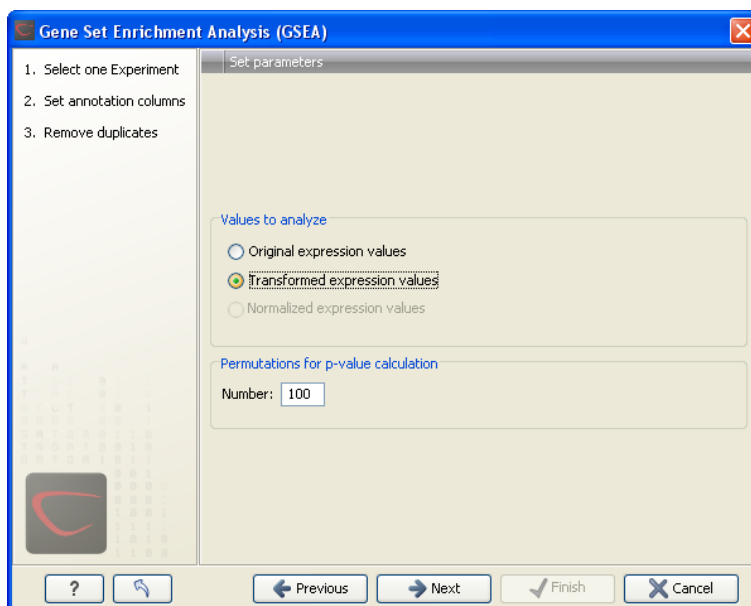
Figure 27.55: *Gene set enrichment analsysis parameters.*

At the top, you can select which values to analyze (see section 27.2.1).

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 15.2). 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 27.56.

The table shows the following information:

- **Category**. This is the identifier for the category.

Figure 27.56: *The result of gene set enrichment analysis on GO biological process.*

- **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.

27.7 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.

27.7.1 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 | Transcriptomics Analysis ()| General Plots | Create Histogram ()

Select a number of samples (() or ()) or a graph track. 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 27.57.
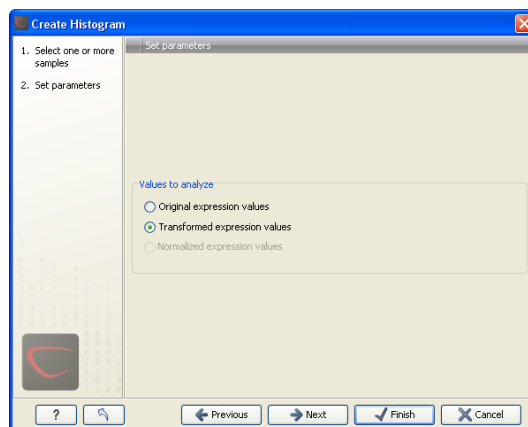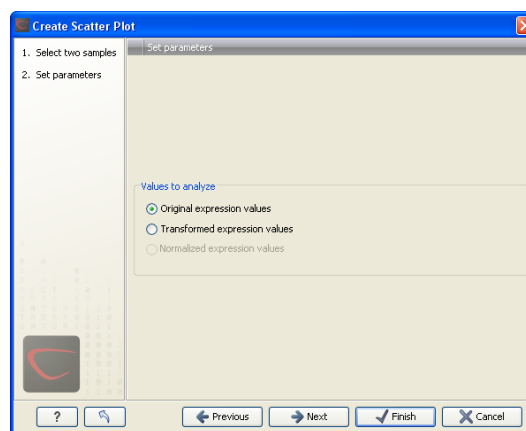
Figure 27.57: *Selcting which values the histogram should be based on.*

In this dialog, you select the values to be used for creating the histogram (see section 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Viewing histograms

The resulting histogram is shown in a figure 27.58
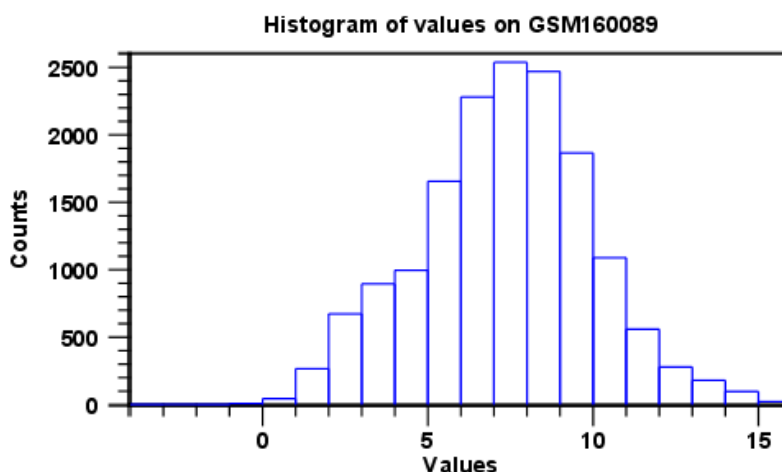
Figure 27.58: *Histogram showing the distribution of transformed expression values.*

The histogram shows the expression value on the x axis (in the case of figure 27.58 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.

- **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 11.6).

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 27.59.

Figure 27.59: *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**

27.7.2 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 | Transcriptomics Analysis ( )| General Plots | Create MA Plot ( )

Select two samples (( ) or ( )). Clicking **Next** will display a dialog as shown in figure 27.60.
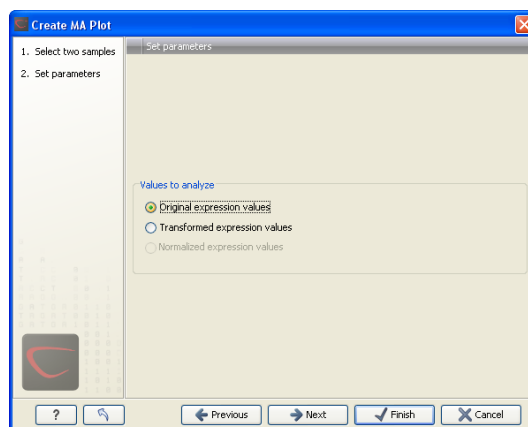
Figure 27.60: *Selcting which values the MA plot should be based on.*

In this dialog, you select the values to be used for creating the MA plot (see section 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

Viewing MA plots

The resulting plot is shown in a figure 27.61.

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 27.61 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 27.2.2).

Figure 27.62 shows the same two samples where the MA plot has been created using log2 transformed 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.
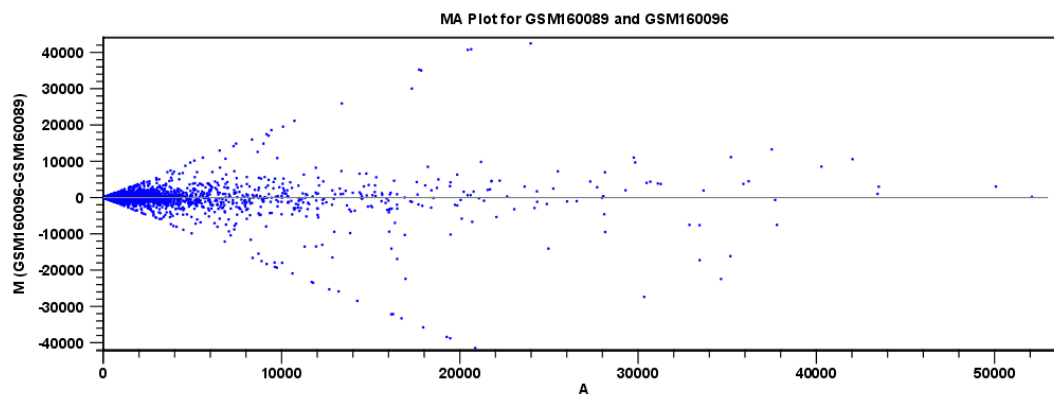
Figure 27.61: *MA plot based on original expression values.*
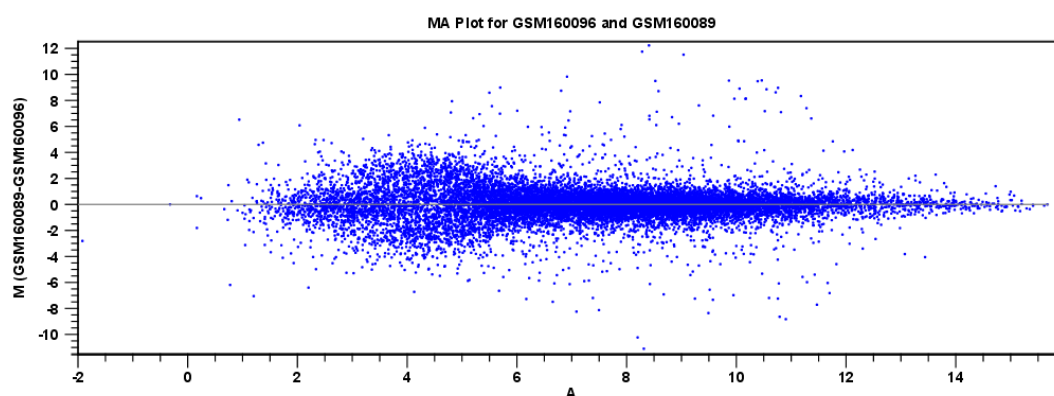
Figure 27.62: *MA plot based on transformed expression values.*

- **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 11.6).

27.7.3   Scatter plot

As described in section 27.1.4, an experiment can be viewed as a scatter plot. However, you can also create a "stand-alone" scatter plot of two samples:

Toolbox | Transcriptomics Analysis ( )| General Plots | Create Scatter Plot ( )

Select two samples (( ) or ( )). Clicking **Next** will display a dialog as shown in figure 27.63.

Figure 27.63: *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 27.2.1).

Click **Next** if you wish to adjust how to handle the results (see section 15.2). If not, click **Finish**.

For more information about the scatter plot view and how to interpret it, please see section 27.1.4.

Chapter 28

Cloning and cutting

Contents

CLC Main 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

533

28.1 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 Main 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 as well as sequences to be used as vector (figure 28.1).
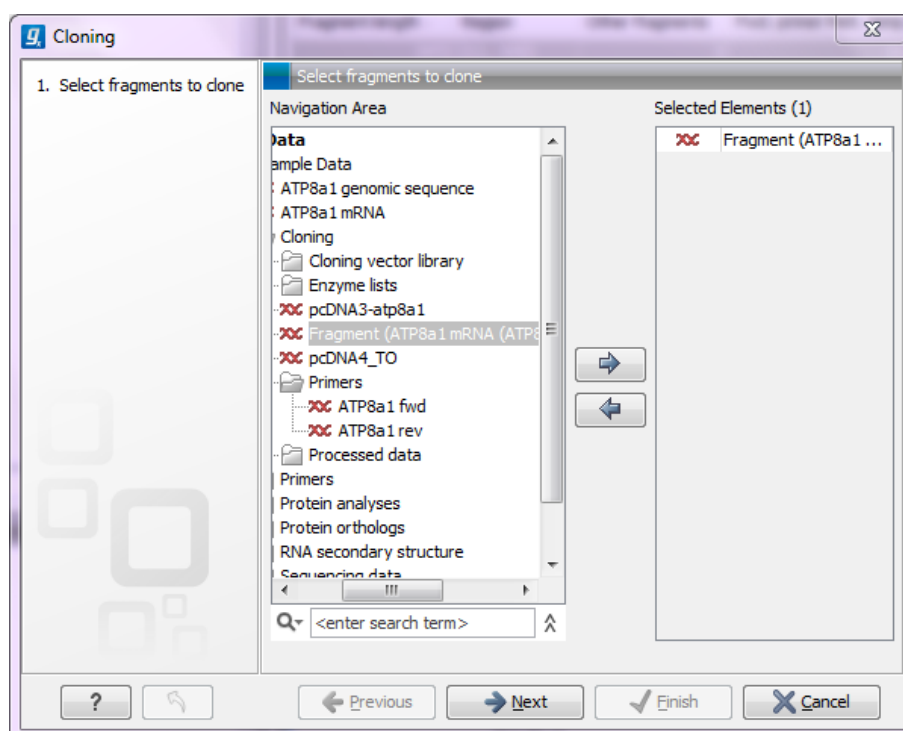
Figure 28.1: *Selecting one or more sequences containing the fragments you want to clone.*

The *CLC Main Workbench* will now create a sequence list of the selected fragments and vector sequences (if you have selected both fragments and vectors) and open it in the cloning editor as shown in figure 28.2.

When you save the cloning experiment, it is saved as a **Sequence list**. See section 17.6 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 view. Just:

right-click anywhere on the empty white area | Add Sequences
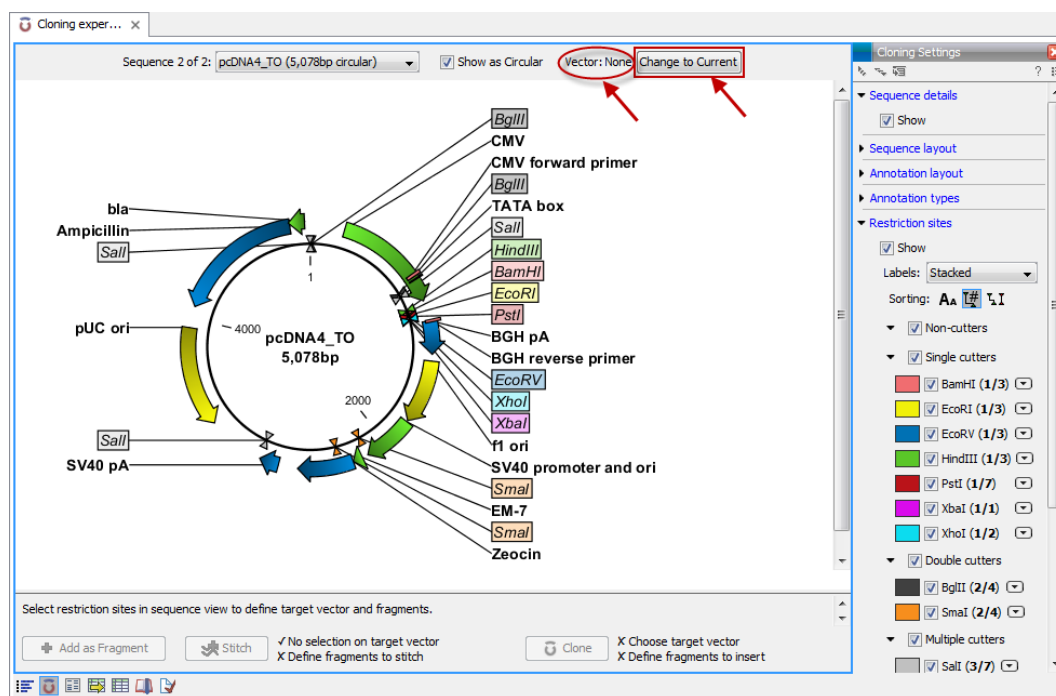
Figure 28.2: *Cloning editor.*

28.1.1 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 17.1 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, you can select whether or not to select a vector. When no vector has been selected a button **Change to Current** is enabled. This button can be used to select the currently shown sequence as **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).

There are essentially three ways of performing cloning in the *CLC Main 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 workflow* 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 Main Workbench* also supports *Gateway cloning* (see section 28.2).

The cloning editor

The cloning editor can be activated in different ways. One way is to click on the **Cloning Editor** icon ( ) in the view area when a sequence list has been opened in the sequence list editor. Another way is to create a new cloning experiment (the actual data object will still be a sequence list) using the **Cloning** ( ) action from the toolbox. Using this action the user collects a set of existing sequences and creates a new sequence list.

The cloning editor can be used in two different ways:

1. **The cloning mode** is utilized when the user has selected one of the sequences as 'Vector'. In the cloning mode, the user opens up the vector by applying one or more cuts to the vector, thereby creating an opening for insertion of other sequence fragments. From the remaining sequences in the cloning experiment/sequence list, either complete sequences or fragments created by cutting can be inserted into the vector. In the cloning adapter dialog, the user can switch the order of the inserted fragments and rotate them prior to adjusting the overhangs to match the cloning conditions.

2. **The stitch mode** is utilized when the user deselects or has not selected a sequence as 'Vector'. In stitch mode, the user can select a number of fragments (either full sequences or cuttings) from the cloning experiment. These fragments can then be stitched together into one single new and longer sequence. In the stitching adapter dialog, the user can switch order and rotate the fragments prior to adjusting the overhangs to match the stitch conditions.

28.1.2   The cloning workflow

The *cloning workflow* is designed to support restriction cloning workflows 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 28.3.1). 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 28.3).

In this example you can see that there are now two options listed in the panel below the view. This is because there are now two 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** ( ).

Figure 28.3: *HindIII and XhoI cut sites selected to cut out 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) (figure 28.4).

Just as when cutting out the fragment, there is a lot 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**.
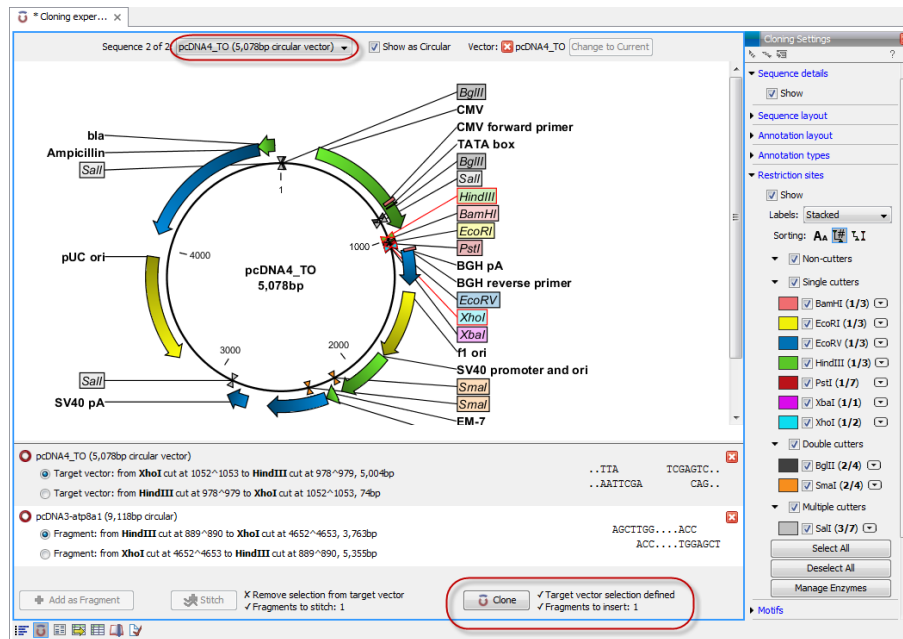
Figure 28.4: *HindIII and XhoI sites used to open the vector. Note that the "Cloning" button has now been enabled as both criteria ("Target vector selection defined" and "Fragments to insert:...") have been defined.*

When the right target vector is selected, you are ready to **Perform Cloning** ( ), see below.

Perform cloning

Once selections have been made for both fragments and vector, click **Cloning** ( ). This will display a dialog to adapt overhangs and change orientation as shown in figure 28.5)
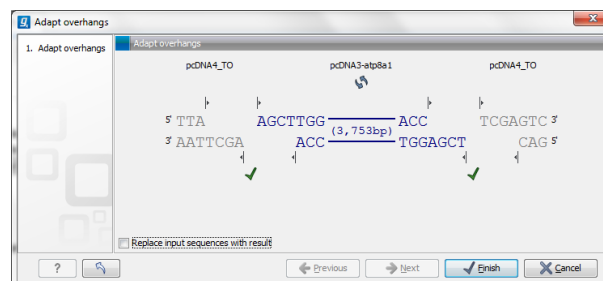
Figure 28.5: *Showing the insertion point of the vector.*

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 (➡)/ (⬅).

There is an option for the result of the cloning: **Replace input sequences with result**. Per default, the construct will be opened in a new view and can be saved separately. By selecting this option, the construct will also be added to the input sequence list and the original fragment and vector sequences will be deleted.

When you click **Finish** the final construct will be shown (see figure 28.6).
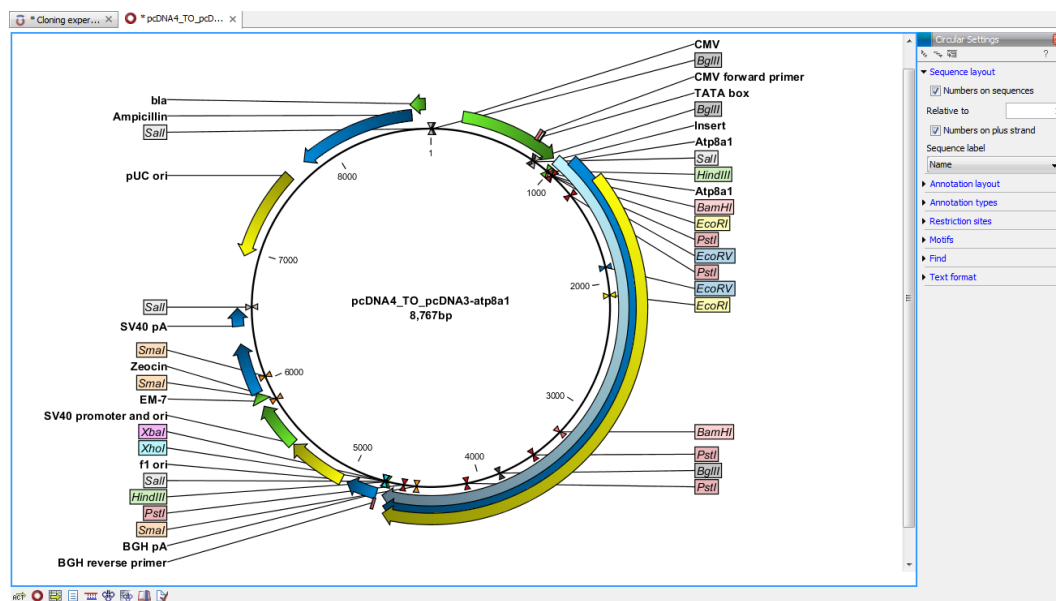
Figure 28.6: *The final construct.*

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.

28.1.3   Manual cloning

If you wish to use the manual way of cloning (as opposed to using the cloning workflow explained above in section 28.1.2), 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 28.7.

- **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:
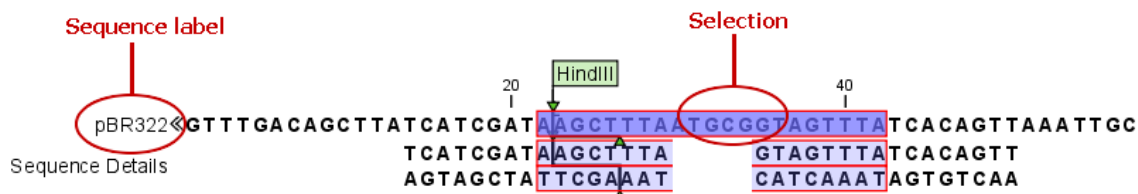
Figure 28.7: *The red circles mark the two places you can use for manipulating the sequences.*

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 28.8).
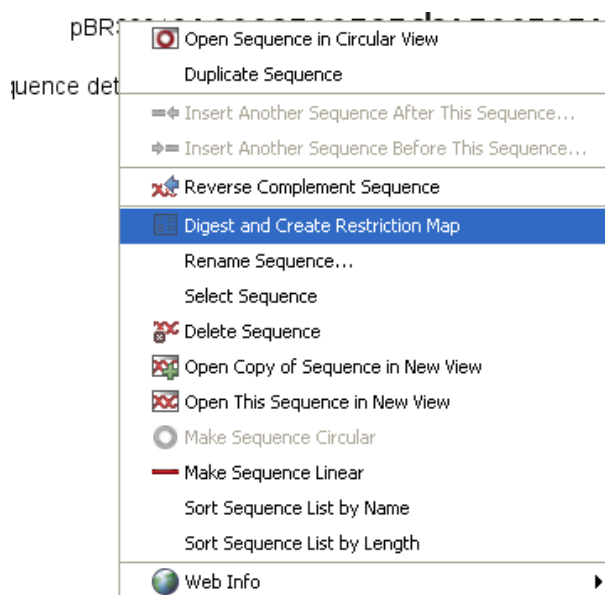
Figure 28.8: *Right click on the sequence in the cloning view.*

- **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 28.4.1

- **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 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 28.9).

- **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**).

- **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.
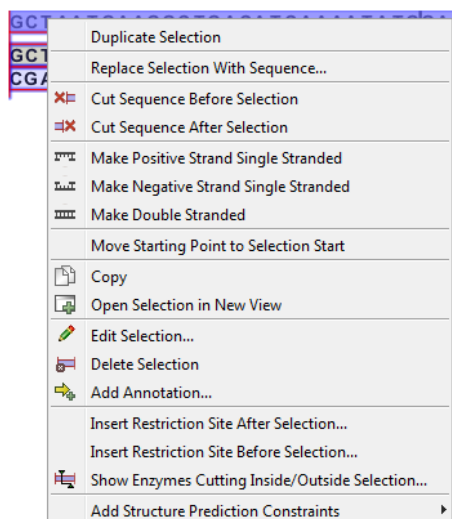
Figure 28.9: *Right click on a sequence selection in the cloning view.*

- **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 (▉).** This will copy the selected region to the clipboard, which will enable it for use in other programs.

- **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.

- **Insert Restriction Sites After/Before Selection.** This will show a dialog where you can choose from a list restriction enzymes (see section 28.1.4).

- **Show Enzymes Cutting Inside/Outside Selection (▉).** This will add enzymes cutting this selection to the Side Panel.

- **Add Structure Prediction Constraints.** This is relevant for RNA secondary structure prediction (see section 31.1.4).

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 28.10).
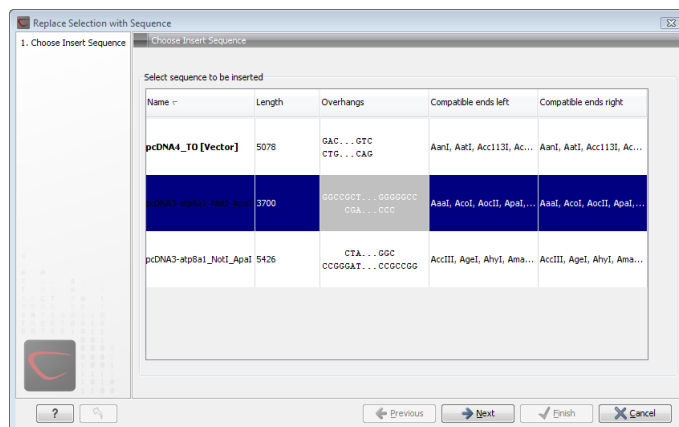
Figure 28.10: *Select a sequence for insertion.*

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 workflow described in section 28.1.2.

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 28.11).
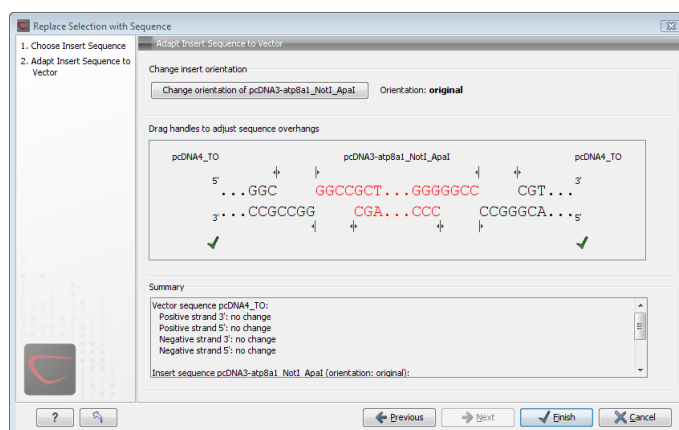
Figure 28.11: *Drag the handles to adjust overhangs.*

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.
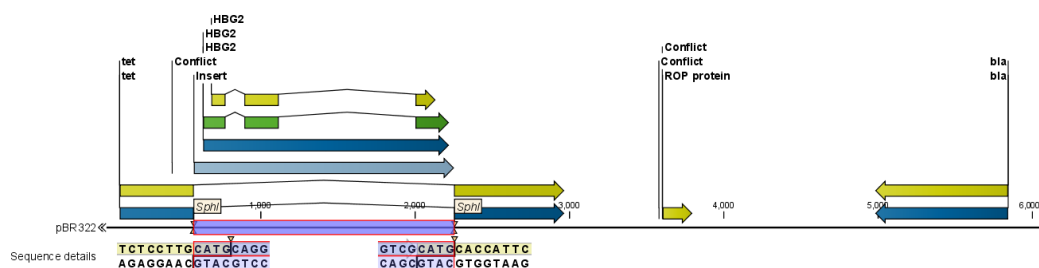
Figure 28.12: *One sequence is now inserted into the cloning vector. The sequence inserted is automatically selected.*

28.1.4   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 28.13

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.

28.2   Gateway cloning

CLC Main Workbench offers tools to perform *in silico* Gateway cloning[2], including Multi-site Gateway cloning.

The three tools for doing Gateway cloning in the *CLC Main Workbench* mimic the procedure followed in the lab:

---

[2]Gateway is a registered trademark of Invitrogen Corporation
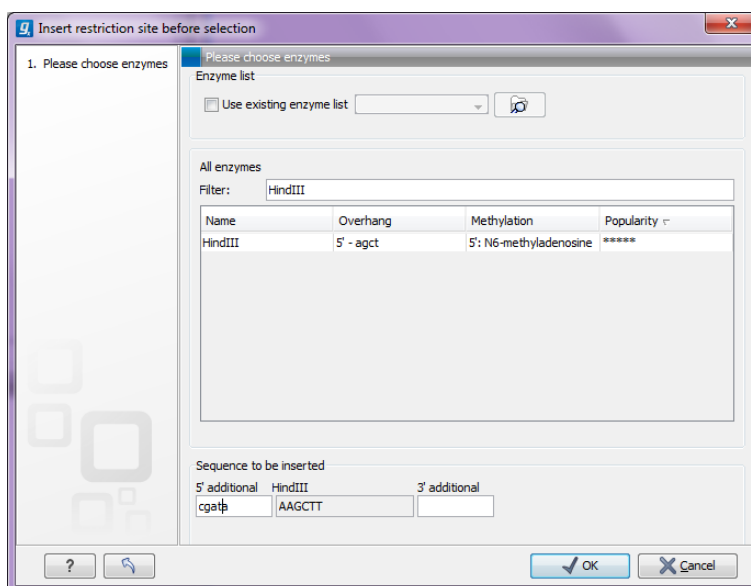
Figure 28.13: *Inserting the HindIII recognition sequence.*

- 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. For Multi-site gateway cloning, multiple entry clones can be created that can recombine in the LR reaction.

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 Main 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`

28.2.1 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 Main 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 or 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 allow you to choose which attB sites you wish to add to each end of the fragment as shown in figure 28.14.
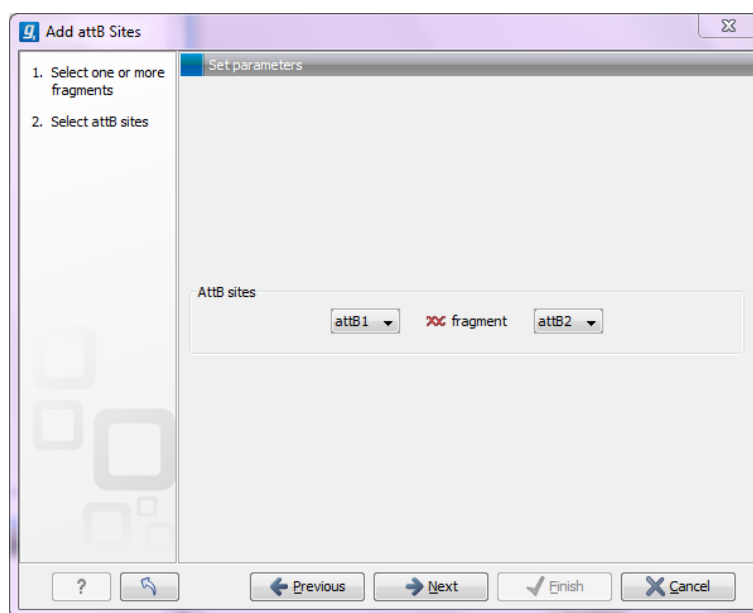
Figure 28.14: *Selecting which attB sites to add.*

The default option is to use the attB1 and attB2 sites. If you have selected several fragments and wish to add different combinations of sites, you will have to run this tool once for each combination.

Click **Next** 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 28.20 where a Shine-Dalgarno site has been added between the attB site and the gene of interest.

At the top of the dialog (see figure 28.15), you can specify primer additions such as a Shine-Dalgarno site, start codon etc. Click in the text field and press **Shift + F1 (Shift + Fn + F1 on Mac)** to show some of the most common additions (see figure 28.16).

Use the up and down arrow keys to select a tag and press **Enter**. This will insert the selected sequence as shown in figure 28.17.

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 28.17).

This default list of primer additions can be modified, see section 28.2.1.
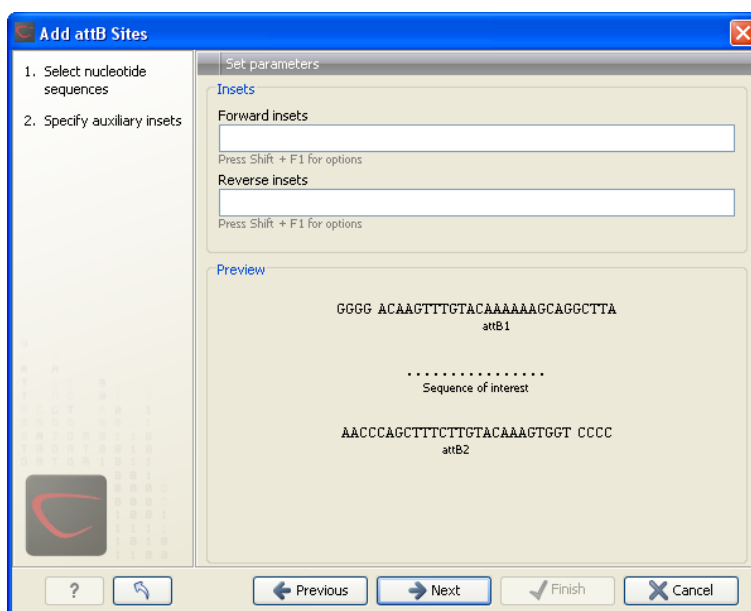
Figure 28.15: *Primer additions 5' of the template-specific part of the primer.*
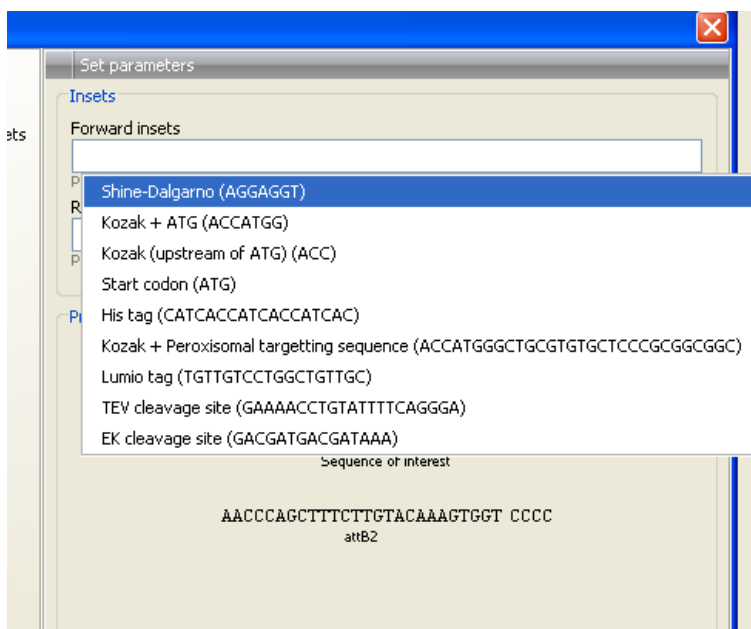
Figure 28.16: *Pressing Shift + F1 shows some of the common additions. This default list can be modified, see section 28.2.1.*

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 28.18.

The *CLC Main 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 can see when clicking **Next** (see figure 28.19).
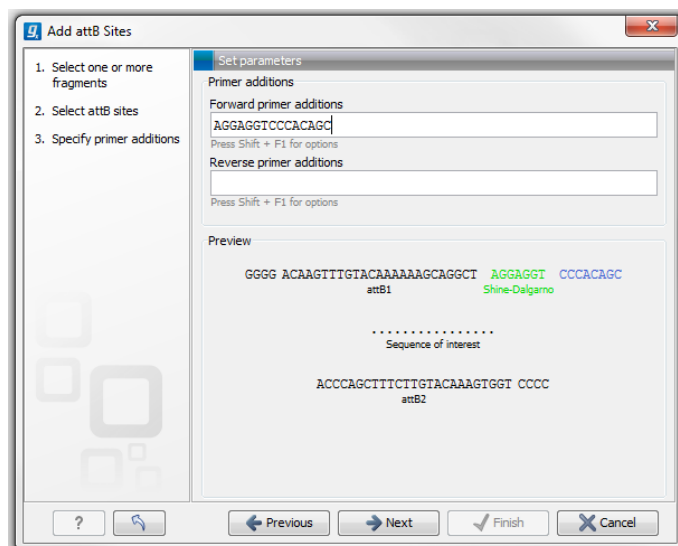
Figure 28.17: *A Shine-Dalgarno sequence has been inserted.*
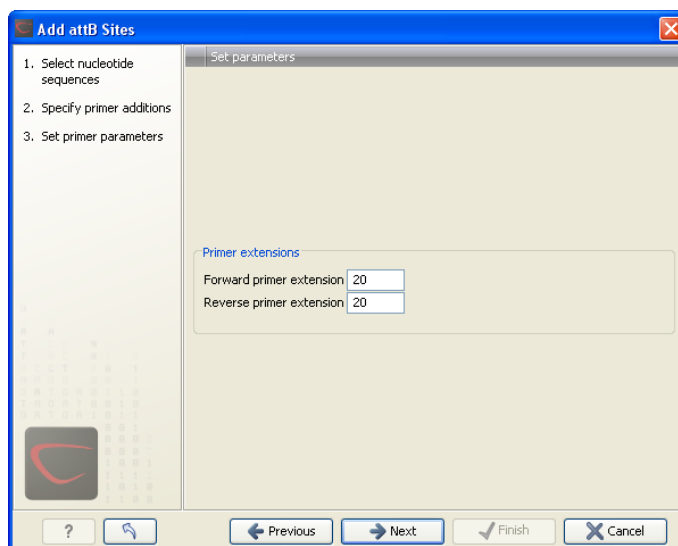
Figure 28.18: *Specifying the length of the template-specific part of the primers.*

Besides the main output which is a copy of 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 15.2). If not, click **Finish**.

The attB sites, the primer additions and the primer regions are annotated in the final result as shown in figure 28.20.

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 28.2.2). 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 28.20.
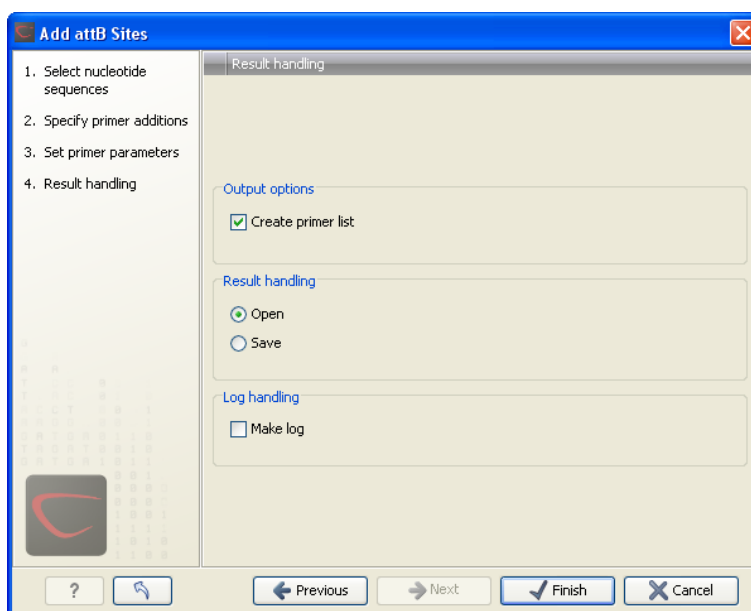
Figure 28.19: *Besides the main output which is a copy of the input sequence(s) now including attB sites and primer additions, you can get a list of primers as output.*
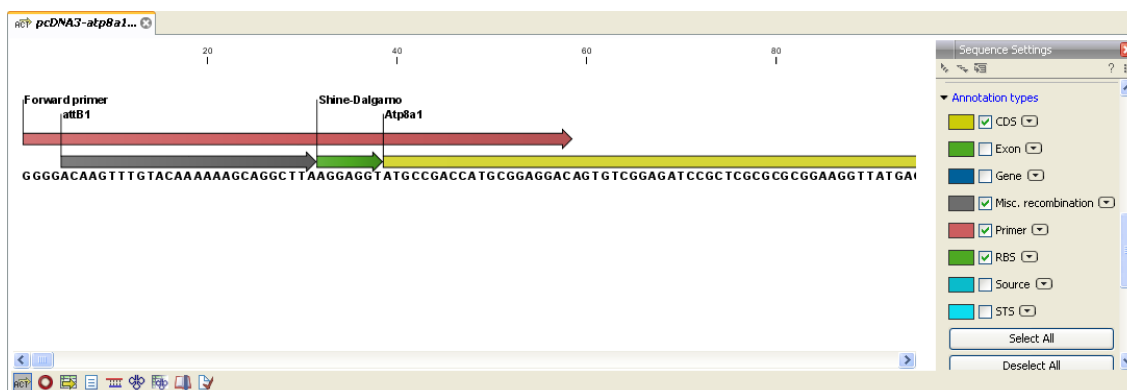
Figure 28.20: *the attB site plus the Shine-Dalgarno primer addition is annotated.*

Extending the pre-defined list of primer additions

The list of primer additions shown when pressing **Shift+F1** (on Mac: Shift + fn + F1) in the dialog shown in figure 28.15 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 28.21).

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).
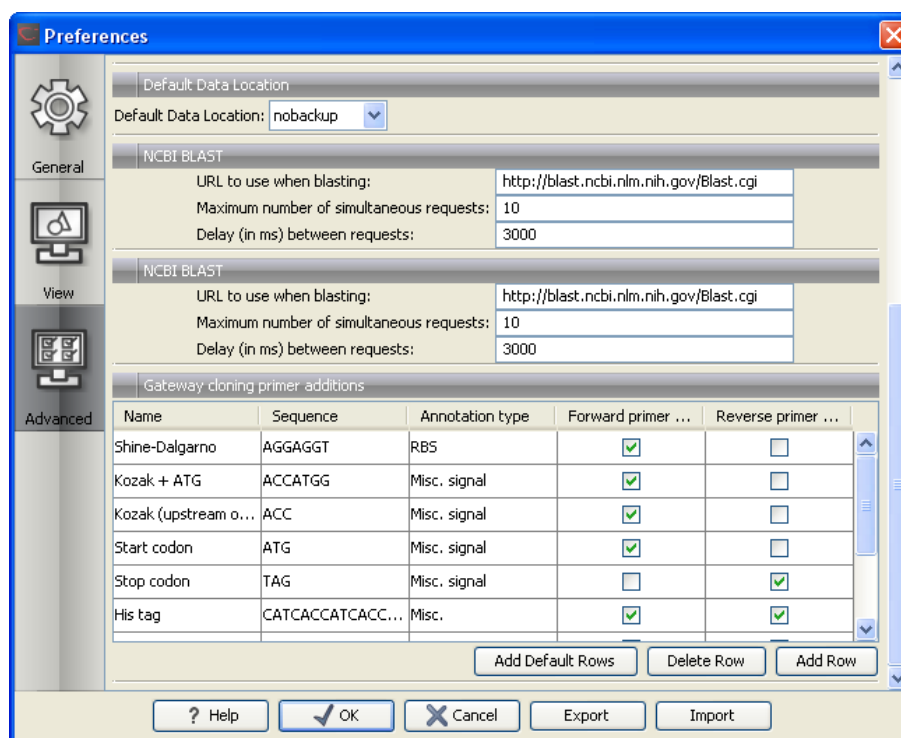
Figure 28.21: *Configuring the list of primer additions available when adding attB sites.*

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.

28.2.2 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 ( )**

This will open a dialog where you can select on or 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 28.2.1). 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 28.22.
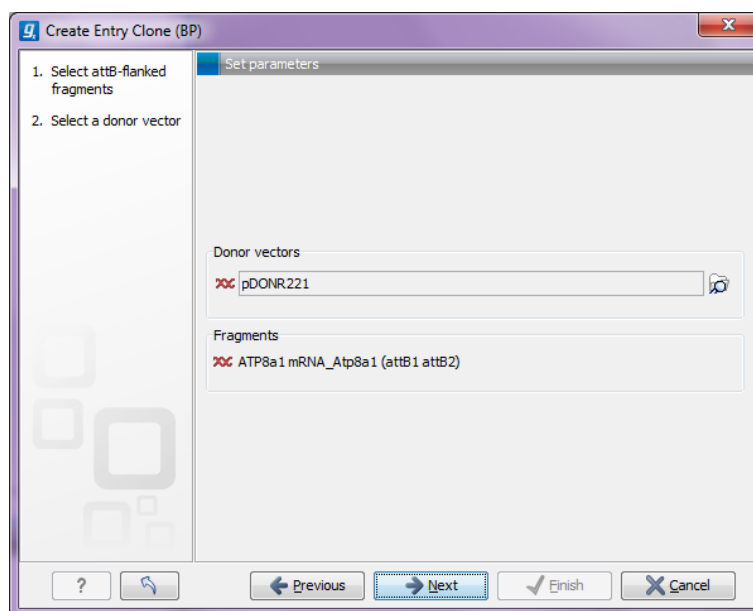
Figure 28.22: *Selecting one or more donor vectors.*

Clicking the **Browse** (image) 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 Main 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 appendix F). Note that the *CLC Main Workbench* only checks that valid attP sites are found - it does not check that they correspond to the attB sites of the selected fragments at this step. If the right combination of attB and attP sites is not found, no entry clones will be produced.

Below there is a preview of the fragments selected and the attB sites that they contain. This can be used to get an overview of which entry clones should be used and check that the right attB sites have been added to the fragments.

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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 28.23.
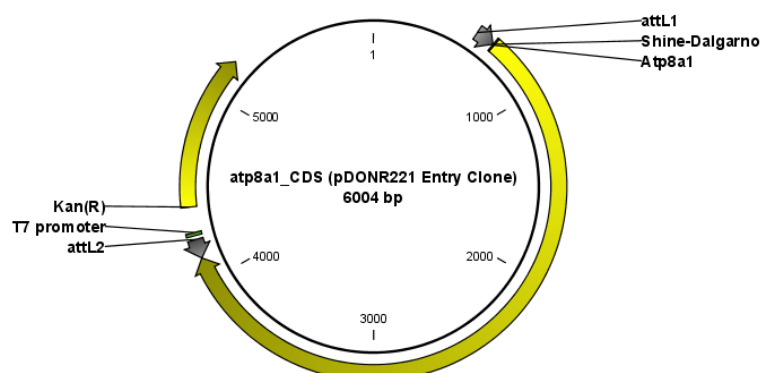
Figure 28.23: *The resulting entry vector opened in a circular view.*

Note that the bi-product of the recombination is not part of the output.

28.2.3   Create expression clones (LR)

The final step in the Gateway cloning work flow is to recombine the entry clone into a destination 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 (** **)**

This will open a dialog where you can select on or more entry clones (see how to create an entry clone in section 28.2.2). If you wish to perform separate LR reactions with multiple entry clones, you should run the **Create Expression Clone** in batch mode (see section 15.1).

When you have selected your entry clone(s), click **Next**.

This will display the dialog shown in figure 28.24.
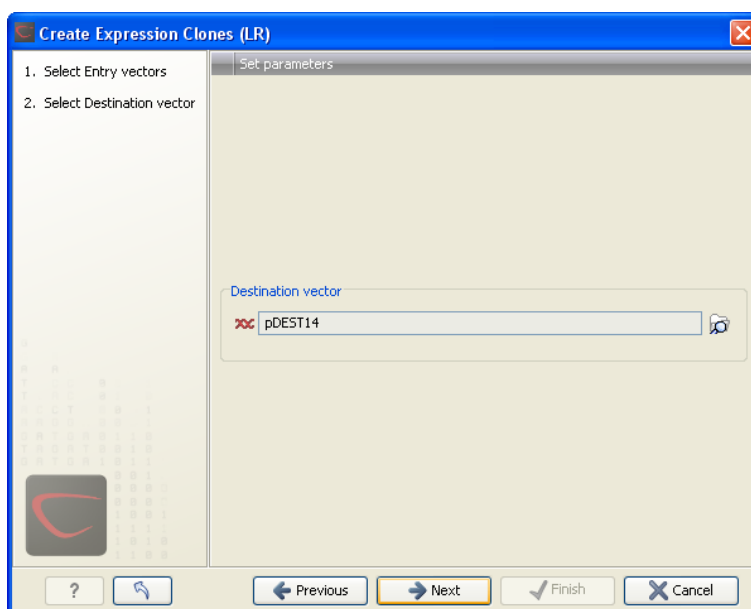
Figure 28.24: *Selecting one or more destination vectors.*

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 Main 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 appendix F). Note that the *CLC Main Workbench* only checks that valid attR sites are found - it does not check that they correspond to the attL sites of the selected fragments at this step. If the right combination of attL and attR sites is not found, no entry clones will be produced.

When performing multi-site gateway cloning, the *CLC Main Workbench* will insert the fragments (contained in entry clones) by matching the sites that are compatible. If the sites have been defined correctly, an expression clone containing all the fragments will be created. You can find an explanation of the multi-site gateway system at `http://tools.invitrogen.com/downloads/gateway-multisite-seminar.html`

Click **Next** if you wish to adjust how to handle the results (see section 15.2). 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 28.25.
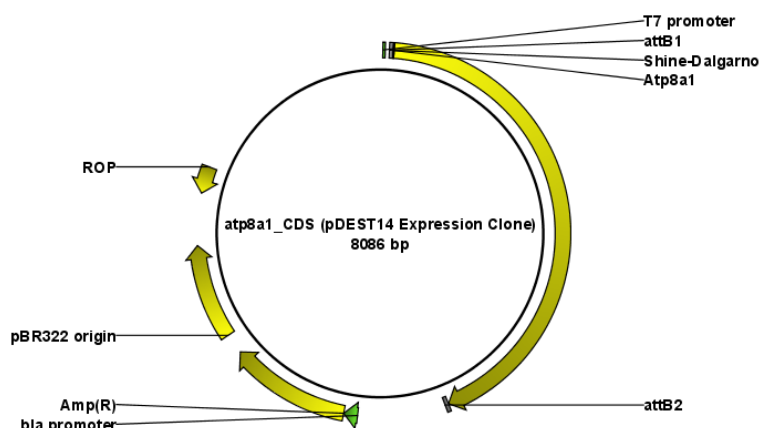
Figure 28.25: *The resulting expression clone opened in a circular view.*

You can choose to create a sequence list with the bi-products as well.

28.3 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 a description of how to do restriction site analyses via the toolbox (you can run more extensive analysis via the restriction analysis tools available in the toolbox). 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.

28.3.1 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 28.26 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.

Figure 28.26: *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 28.27). 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 28.29).

- **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 28.28).

Figure 28.27: *Restriction site labels shown as flags.*

Note that in a circular view, the **Stacked** and **Radial** options also affect the layout of annotations.
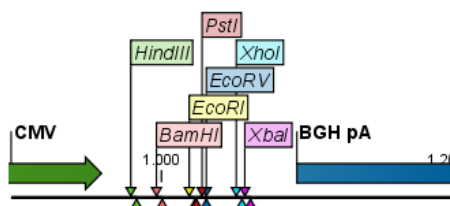
Figure 28.28: *Restriction site labels stacked.*
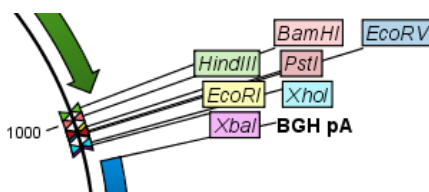
Figure 28.29: *Restriction site labels in radial layout.*

Sort enzymes

Just above the list of enzymes there are three buttons to be used for sorting the list (see figure 28.30):
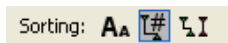
Figure 28.30: *Buttons to sort restriction enzymes.*

- **Sort enzymes alphabetically** (**A**A).   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** (⌐⌐). 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.
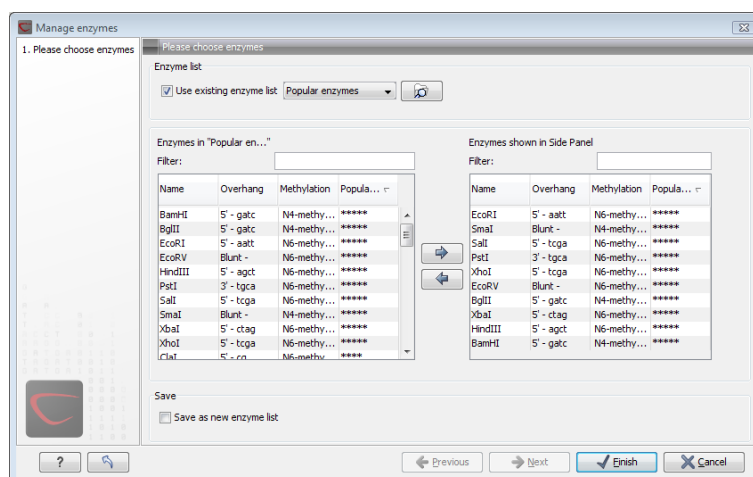
Figure 28.31: *Adding or removing enzymes from the Side Panel.*

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 28.31.

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 28.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you can see all the enzymes that are in the list selected above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [3].

- To the **right**, you can see the 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 28.50.
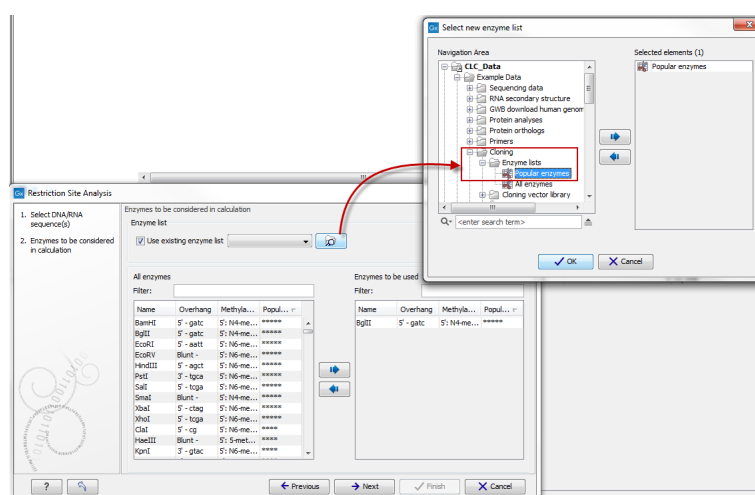
---

[3]The *CLC Main 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 E
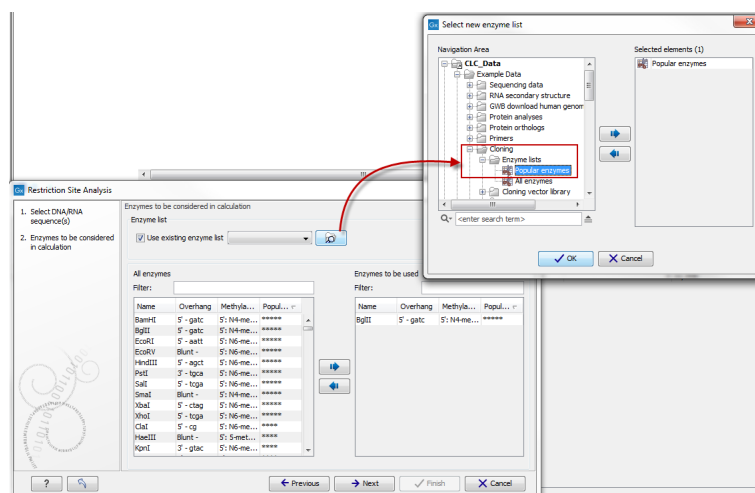
Figure 28.32: *Selecting enzymes.*

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 28.51), or use the view of enzyme lists (see 28.5).
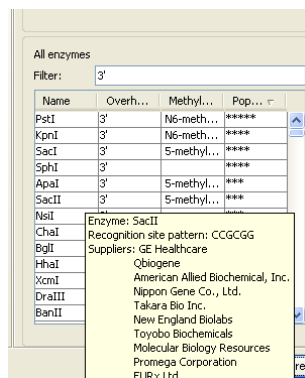
Figure 28.33: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

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 11.6) for future use.

Show enzymes cutting inside/outside selection

Section 28.3.1 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 28.34 where you can specify which enzymes should initially be considered.
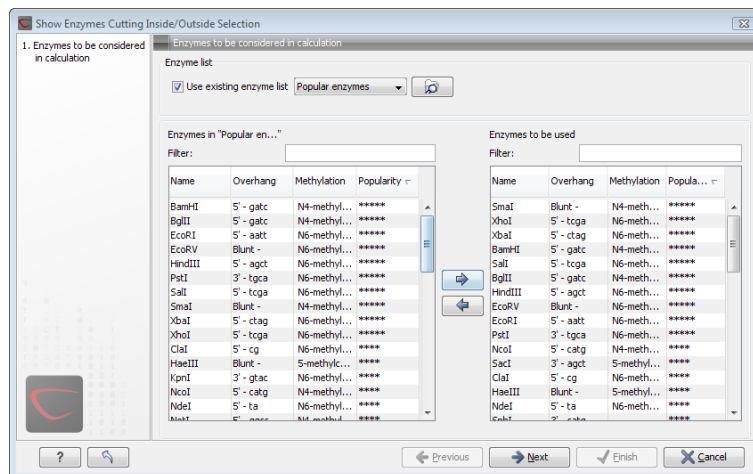
Figure 28.34: *Choosing enzymes to be considered.*

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 28.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you can see all the enzymes that are in the list selected above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [4].

- To the **right**, you can see the 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 28.50.

---

[4]The *CLC Main 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 E
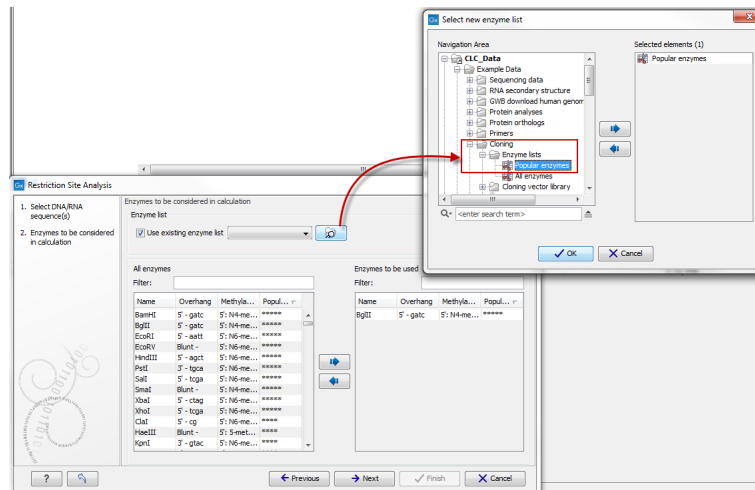
Figure 28.35: *Selecting enzymes.*

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 28.51), or use the view of enzyme lists (see 28.5).
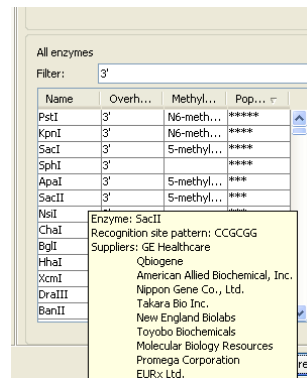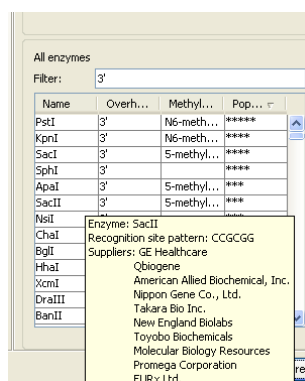

Figure 28.36: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

Clicking **Next** will show the dialog in figure 28.37.
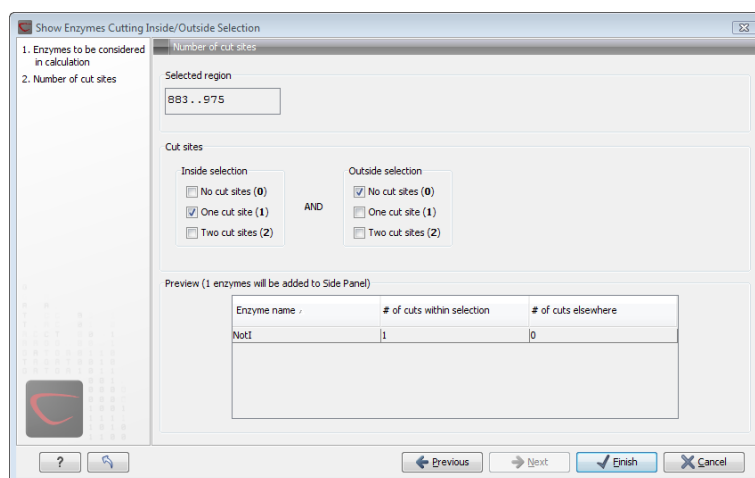
Figure 28.37: *Deciding number of cut sites inside and outside the selection.*

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:

<p style="text-align:center">right-click the restriction site | Show Enzymes with Compatible Ends (⌐I)</p>

This will display the dialog shown in figure 28.38.
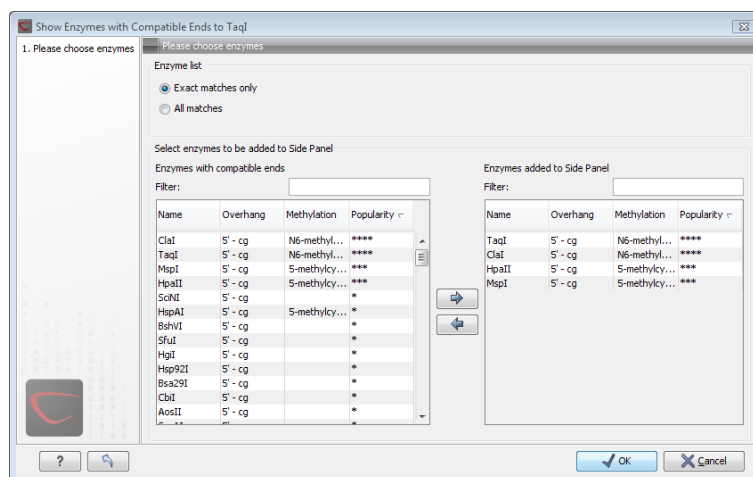
Figure 28.38: *Enzymes with compatible ends.*

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.

28.3.2 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:

$\qquad$ **Toolbox | Cloning and Restriction Sites (🖼️)| Restriction Site Analysis (✂️)**

This will display the dialog shown in figure 28.39.
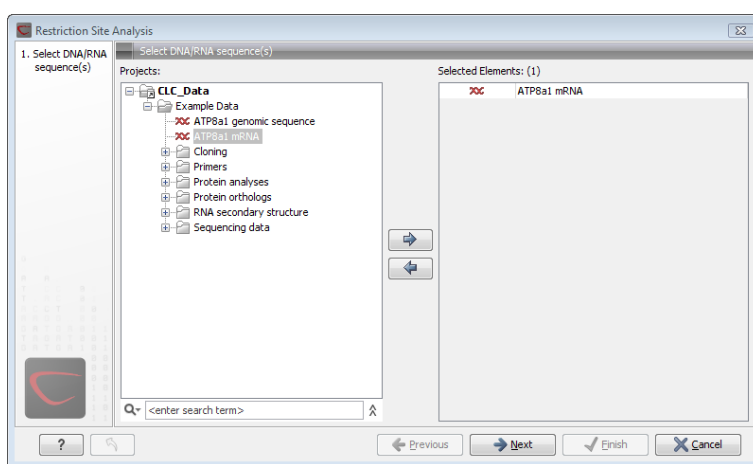
Figure 28.39: *Choosing sequence ATP8a1 mRNA for restriction map analysis.*

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 28.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you can see all the enzymes that are in the list selected above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [5].

---

[5]The *CLC Main 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 E

- To the **right**, you can see the 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 28.50.

Figure 28.40: *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 28.51), or use the view of enzyme lists (see 28.5).

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 28.42.

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**)

Figure 28.41: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*
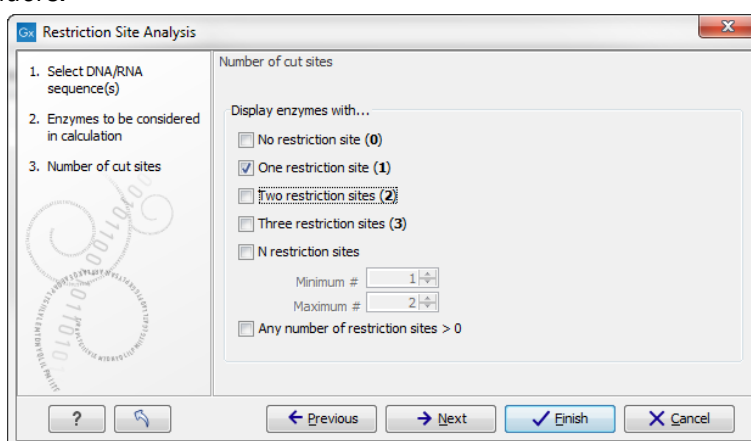
Figure 28.42: *Selecting number of cut sites.*

- 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 28.43.

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 28.44) and save the annotations for later use.
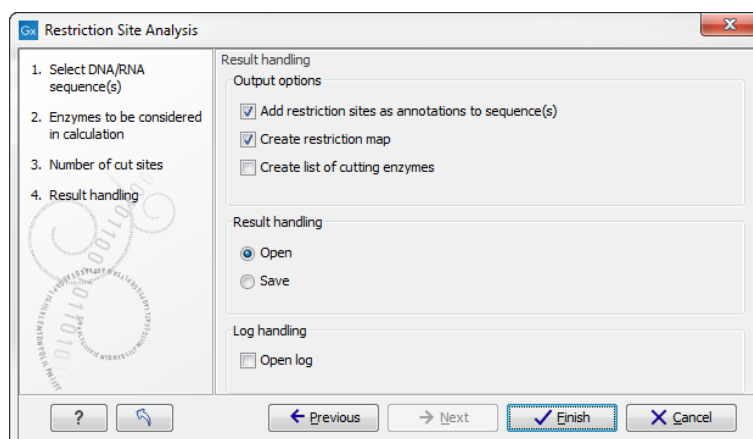
Figure 28.43: *Choosing to add restriction sites as annotations or creating a restriction map.*

- **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 28.45. 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 figure28.46).
  - As a **virtual gel** simulation which shows the fragments as bands on a gel (see figure 28.48).
    For more information about gel electrophoresis, see section 28.4.

The following sections will describe these output formats in more detail.

In order to complete the analysis click **Finish** (see section 15.2 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 28.44.   See section 17.3 for more information about viewing

Figure 28.44: *The result of the restriction analysis shown as annotations.*

annotations.

Table of restriction sites

The restriction map can be shown as a table of restriction sites (see figure 28.45).

Each row in the table represents a restriction enzyme. The following information is available for each enzyme:

Figure 28.45: *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 28.45 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 28.46.

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

Figure 28.46: *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 28.4.1.

28.4   Gel electrophoresis

CLC Main 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.

28.4.1   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 **??**.

- 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 17.2).
    - Ordinary sequence view (see section 17.1).
    - Graphical view of sequence lists (see section 17.6).
    - Cloning editor (see section 28.1).
    - Primer designer (see section 25.3).

    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 17.1.2.

The view of the gel is explained in section 28.4.3

28.4.2   Separate sequences on gel

To separate sequences without restriction enzyme digestion, first create a sequence list of the sequences in question (see section 17.6). 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.

28.4.3   Gel view

In figure 28.48 you can see a simulation of a gel with its **Side Panel** to the right. This view will be explained in this section.
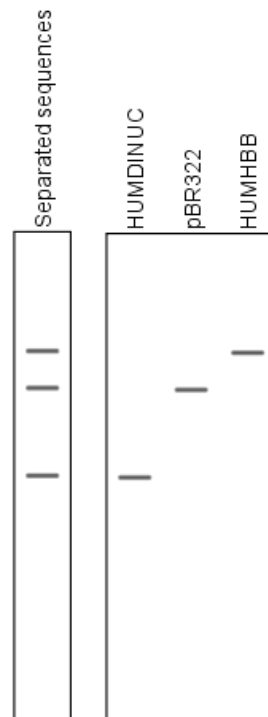
Figure 28.47: *A sequence list shown as a gel.*
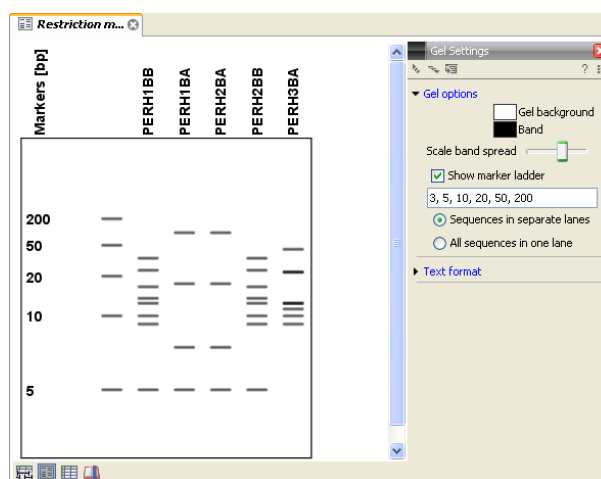
Figure 28.48: *Five lanes showing fragments of five sequences cut with restriction enzymes.*

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**.

28.5 Restriction enzyme lists

CLC Main 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 Main Workbench*.

This section describes how you can create an enzyme list, and how you can modify it.

28.5.1 Create enzyme list

CLC Main 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 E
[8]You can customize the enzyme database for your installation, see section E

File | New | Enzyme list (▦)

This opens the dialog shown in figure 28.49
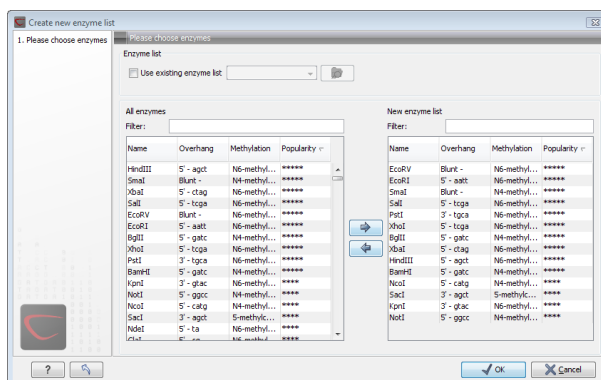
Figure 28.49: *Choosing enzymes for the new enzyme list.*

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 28.5 for more about creating and modifying enzyme lists.

Below there are two panels:

- To the **left**, you can see all the enzymes that are in the list selected above. If you have not chosen to use an existing enzyme list, this panel shows all the enzymes available [9].

- To the **right**, you can see the 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 28.50.

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 28.51), or use the view of enzyme lists (see 28.5).

Click **Finish** to open the enzyme list.

---

[9]The *CLC Main 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 E
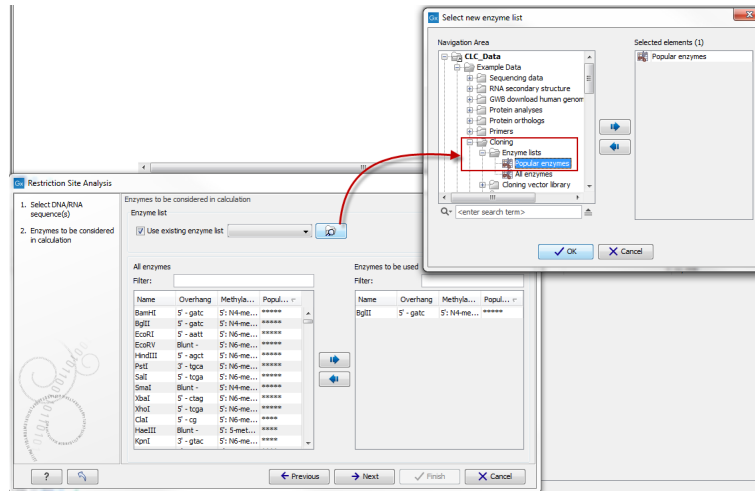
Figure 28.50: *Selecting enzymes.*
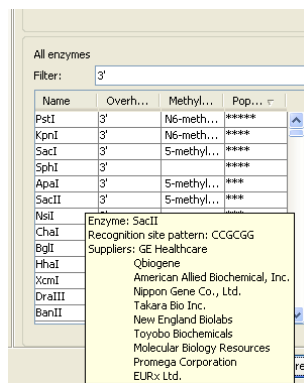
Figure 28.51: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

28.5.2   View and modify enzyme list

An enzyme list is shown in figure 28.52.   The list can be sorted by clicking the columns,
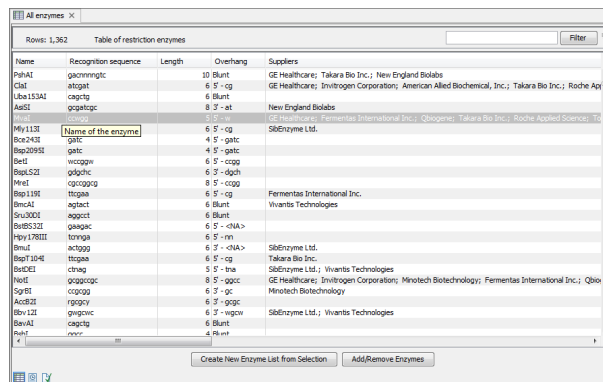
Figure 28.52: *An enzyme list.*

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 28.49 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 29

Sequence alignment

Contents

CLC Main Workbench can align nucleotides and proteins using a *progressive alignment* algorithm (see section 29.6 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.

29.1 Create an alignment

Alignments can be created from sequences, sequence lists (see section 17.6), existing alignments and from any combination of the three.

To create an alignment in *CLC Main 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 29.1.
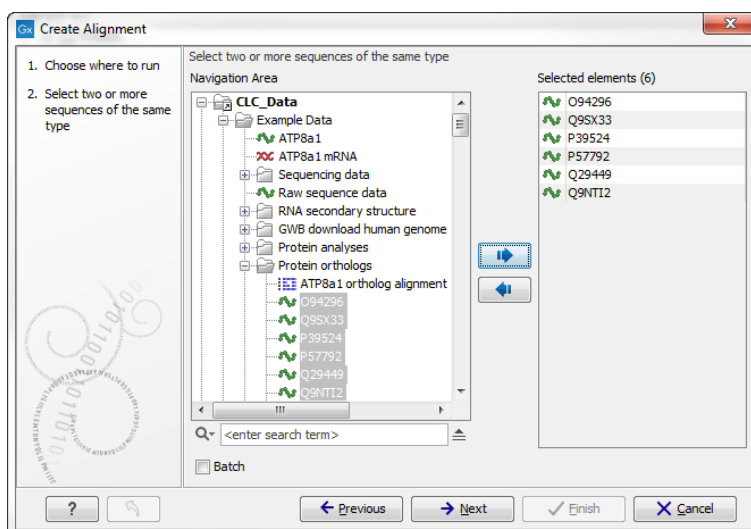
Figure 29.1: *Creating an alignment.*

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 29.2.
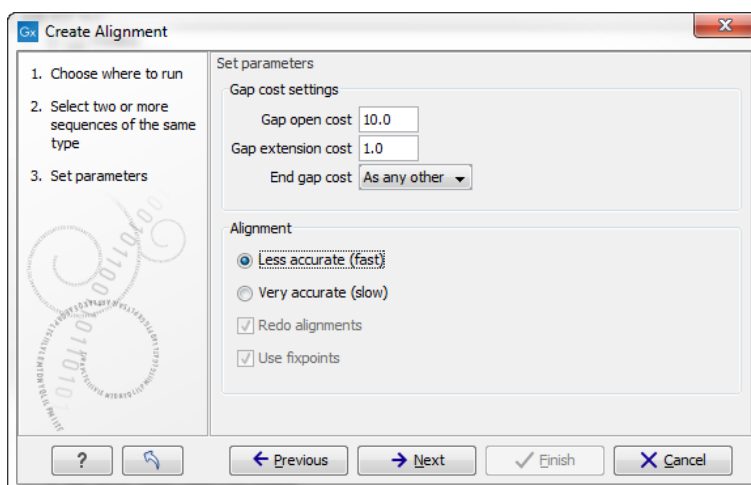
Figure 29.2: *Adjusting alignment algorithm parameters.*

29.1.1   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 Main 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 29.3 and 29.4 illustrate the differences between the different gap scores at the sequence ends.

29.1.2   Fast or accurate alignment algorithm

CLC Main 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.
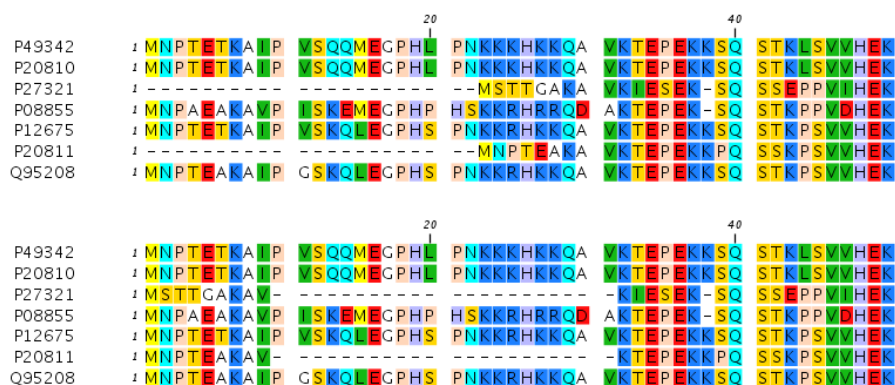
Figure 29.3: *The first 50 positions of two different alignments of seven calpastatin sequences. The top alignment is made with cheap end gaps, while the bottom alignment is made with end gaps having the same price as any other gaps. In this case it seems that the latter scoring scheme gives the best result.*
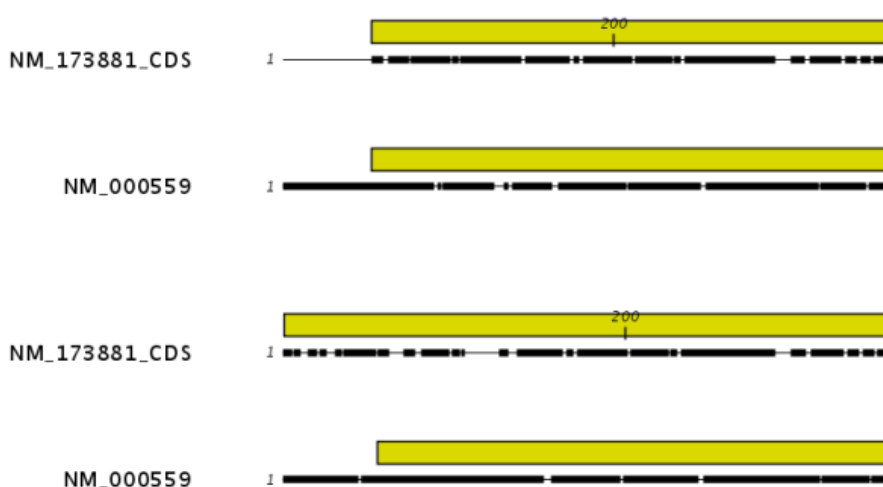
Figure 29.4: *The alignment of the coding sequence of bovine myoglobin with the full mRNA of human gamma globin. The top alignment is made with free end gaps, while the bottom alignment is made with end gaps treated as any other. The yellow annotation is the coding sequence in both sequences. It is evident 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.

29.1.3 Aligning alignments

If you have selected an existing alignment in the first step (29.1), 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 29.5.
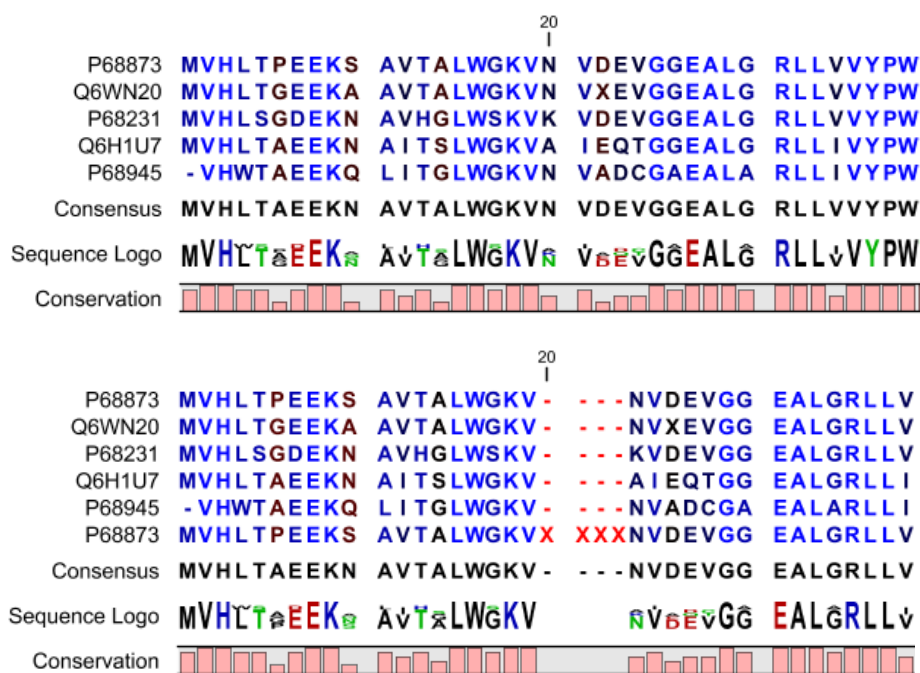
Figure 29.5: *The top figures shows the original alignment. In the bottom panel a single sequence with four inserted X's are aligned to the original alignment. This introduces gaps in all sequences of the original alignment. All other positions in the original alignment are fixed.*

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.

29.1.4 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 29.6). 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 29.7 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
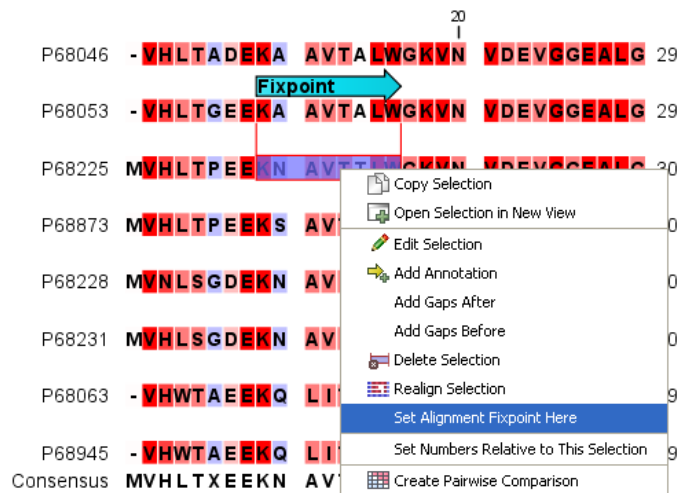
Figure 29.6: *Adding a fixpoint to a sequence in an existing alignment. At the top you can see a fixpoint that has already been added.*
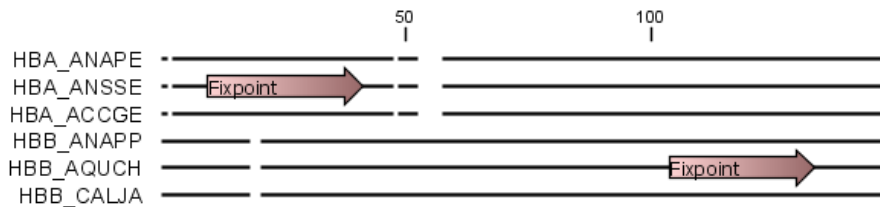
Figure 29.7: *Realigning using fixpoints. In the top view, fixpoints have been added to two of the sequences. In the view below, the alignment has been realigned using the fixpoints. The three top sequences are very similar, and therefore they follow the one sequence (number two from the top) that has 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**

29.2   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 17.1 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 H and G.) Please note that the IUPAC codes are only available for nucleotide alignments.
  - **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 no 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 13.4.
  - ∗ **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 13.4).
    - ∗ **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 29.2.1 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.

29.2.1   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 29.8). 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 29.8). 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 29.8 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.
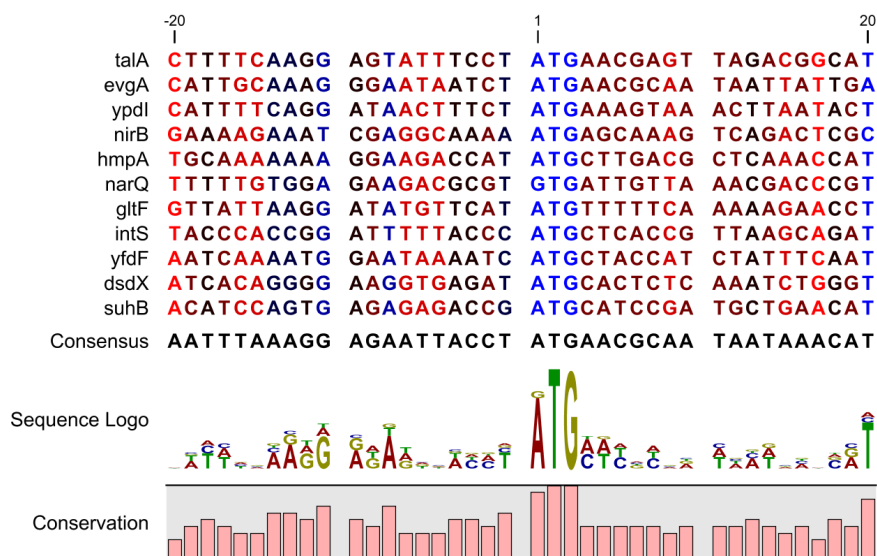
Figure 29.8: *Ungapped sequence alignment of eleven* E. coli *sequences defining a start codon. The start codons start at position 1. Below the alignment is shown the corresponding sequence logo. As seen, a GTG start codon and the usual ATG start codons are present in the alignment. This can also be 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]

29.3   Edit alignments

29.3.1   Move residues and gaps

The placement of gaps in the alignment can be changed by modifying the parameters when creating the alignment (see section 29.1). 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 29.9).

Note! Residues can only be moved when they are next to a gap.

Figure 29.9: *Moving a part of an alignment. Notice the change of mouse pointer to a horizontal arrow.*

29.3.2   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.

29.3.3   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.

In order to delete entire columns:

> **manually select the columns to delete | right-click the selection | click 'Delete Selection'**

29.3.4   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.

If you wish to copy all annotations on the sequence, click the **Copy All Annotations to other Sequences**.

29.3.5   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.

If you have one particular sequence that you would like to use as a reference sequence, it can be useful to move this to the top. This can be done manually, but it can also be done automatically:

> **Right-click the name of a sequence | Move Sequence to Top**

The sequences can also be sorted by similarity, grouping similar sequences together:

> **Right-click the name of a sequence | Sort Sequences by Similarity**

29.3.6   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 29.1).

The same procedure can be used for joining two alignments.

29.3.7   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 29.1).

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.

29.4 Join alignments

CLC Main 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 29.10.
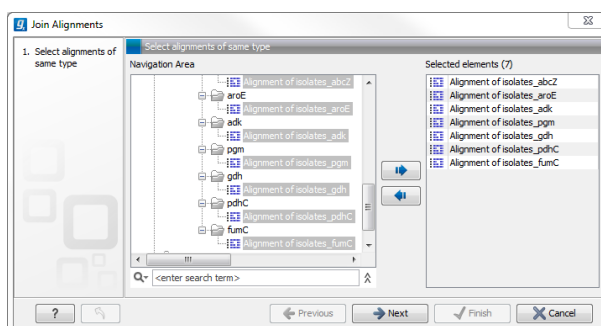
Figure 29.10: *Selecting two alignments to be joined.*

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. In this example seven alignments are selected. Each alignment represents one gene that have been sequenced from five different bacterial isolates from the genus Nisseria. Clicking **Next** opens the dialog shown in figure 29.11.
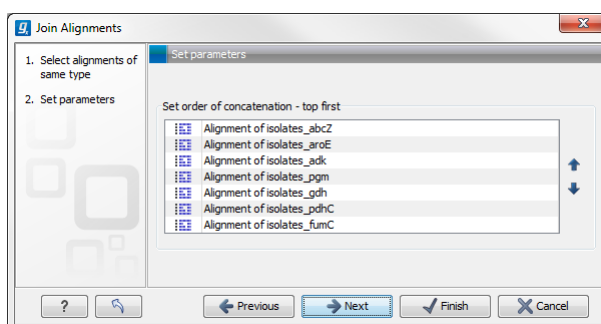
Figure 29.11: *Selecting order of concatenation.*

To adjust the order of concatenation, click the name of one of the alignments, and move it up or down using the arrow buttons.

The result is seen in the lower part of figure 29.12.
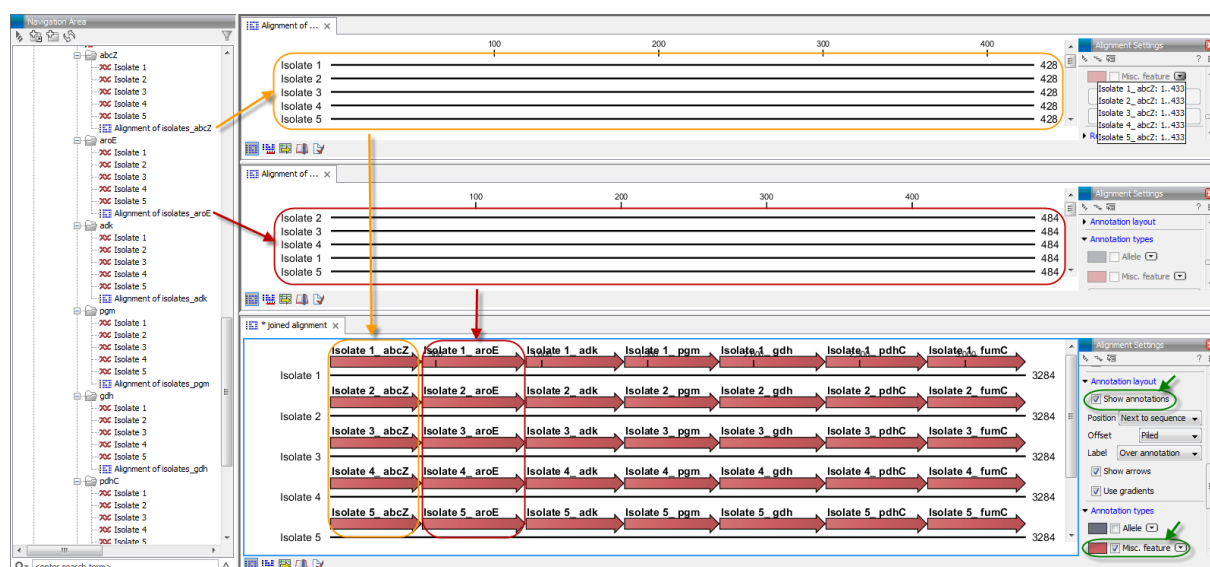
Figure 29.12: *The upper part of the figure shows two of the seven alignments for the genes "abcZ" and "aroE" respectively. Each alignment consists of sequences from one gene from five different isolates. The lower part of the figure shows the result of "Join Alignments". Seven genes have been joined to an artificial gene fusion, which can be useful for construction of phylogenetic trees in cases where only fractions of a genome is available. Joining of the alignments results in one row for each isolate consisting of seven fused genes. Each fused gene sequence corresponds to the number of uniquely named sequences in the joined alignments.*

29.4.1   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 same origin and are thus joined. Consider the joining of the alignments shown in figure 29.12 "Alignment of isolates_abcZ", "Alignment of isolates_aroE", "Alignment of isolates_adk" etc. If a sequence with the same name is found in the different alignments (in this case the name of the isolates: Isolate 1, Isolate 2, Isolate 3, Isolate 4, and Isolate 5), a joined alignment will exist for each sequence name. In the joined alignment the selected alignments will be fused with each other in the order they were selected (in this case the seven different genes from the five bacterial isolates). Note that annotations have been added to each individual sequence before aligning the isolates for one gene at the time in order to make it clear which sequences were fused to each other.

29.5   Pairwise comparison

For a given set of aligned sequences it is possible to 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 Main Workbench* this is done by creating a comparison table:

> **Toolbox in the Menu Bar** | **Alignments and Trees (**⊞**)| Create Pairwise Comparison (**⊞**)**

> or **right-click alignment in Navigation Area | Toolbox | Alignments and Trees (📂)|
> Create Pairwise Comparison (⊞)**

This opens the dialog displayed in figure 29.13:
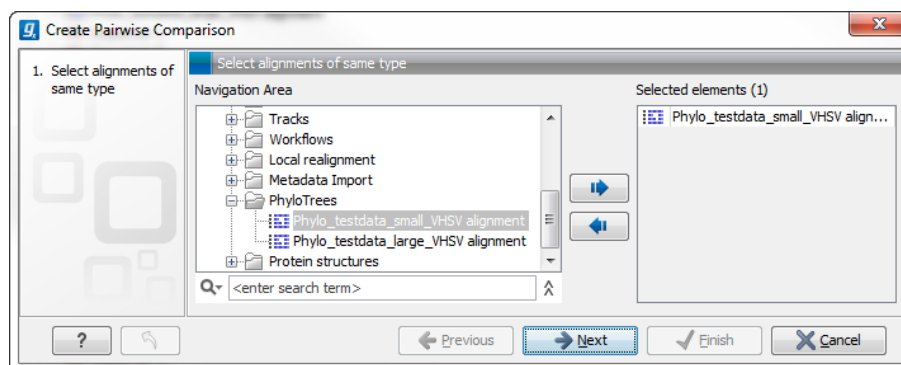
Figure 29.13: *Creating a pairwise comparison table.*

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.

29.5.1   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.

29.5.2   Pairwise comparison parameters

There are five kinds of comparison that can be made between the sequences in the alignment,
as shown in figure 29.14.
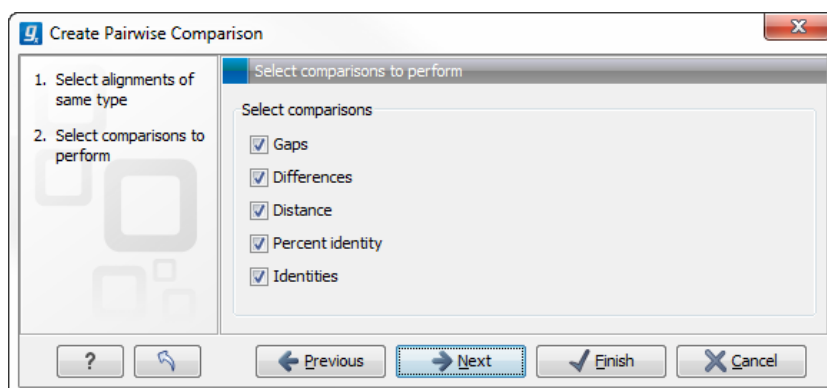
Figure 29.14: *Adjusting parameters for pairwise comparison.*

- **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.

29.5.3 The pairwise comparison table

The table shows the results of selected comparisons (see an example in figure 29.15). 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.
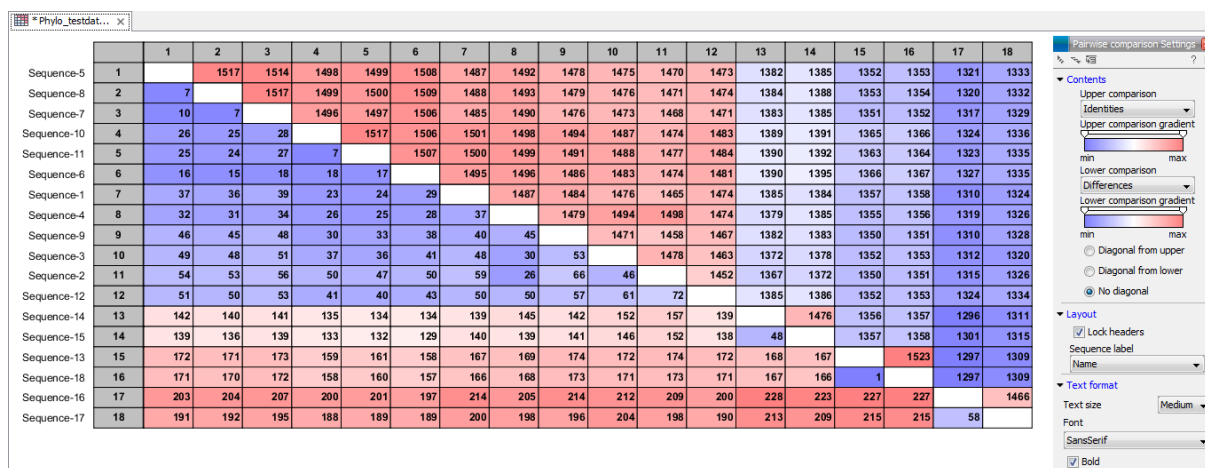
Figure 29.15: *A pairwise comparison table.*

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 Diagona.** 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.

29.6 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 29.16) 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.

29.6.1 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.

Figure 29.16: *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.*

29.6.2 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.

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

All CLC bio's scientific articles are licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. You are free to copy, distribute, display, and use the work for educational purposes, under the following conditions: You must attribute the work in its original form and

Chapter 30

Phylogenetic trees

Contents

30.1 Phylogenetic tree features

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. The focus of this module is the reconstruction and visualization of phylogenetic trees. Phylogenetic trees illustrate the inferred evolutionary history of a set of organisms, and makes it possible to e.g. identify groups of closely related organisms and observe clustering of organisms with common traits. See 30.2.5 for a more detailed introduction to phylogenetic trees.

The viewer for visualizing and working with phylogenetic trees allows the user to create high-quality, publication-ready figures of phylogenetic trees. Large trees can be explored in two alternative tree layouts; circular and radial. The viewer supports importing, editing and visualization of metadata associated with nodes in phylogenetic trees.

Below is an overview of the main features of the phylogenetic tree editor. Further details can be found in the subsequent sections.

Main features of the phylogenetic tree editor:

- Circular and radial layouts.

- Import of metadata in Excel and CSV format.

- Tabular view of metadata with support for editing.

- Options for collapsing nodes based on bootstrap values.

- Re-ordering of tree nodes.

- Legends describing metadata.

- Visualization of metadata though e.g. node color, node shape, branch color, etc.

- Minimap navigation.

- Coloring and labeling of subtrees.

- Curved edges.

- Editable node sizes and line width.

- Intelligent visualization of overlapping labels and nodes.

30.2   Create Trees

For a given set of aligned sequences (see section 29.1) it is possible to infer their evolutionary relationships. In *CLC Main Workbench* this may be done either by using a distance based method or by using maximum likelihood (ML) estimation, which is a statistical approach (see "Bioinformatics explained" in section 30.2.5). Both approaches generate a phylogenetic tree.

Three tools are available for generating phylogenetic trees:

- **K-mer Based Tree Construction** () Is a distance-based method that can create trees based on multiple single sequences. K-mers are used to compute distance matrices for distance-based phylogenetic reconstruction tools such as neighbor joining and UPGMA (see section 30.2.5). This method is less precise than the "Create Tree" tool but it can cope with a very large number of long sequences as it does not require a multiple alignment.

The k-mer based tree construction tool is especially useful for whole genome phylogenetic reconstruction where the genomes are closely releated, i.e. they differ mainly by SNPs and contain no or few structural variations.

- **Maximum Likelihood Phylogeny** (⚏) The most advanced and time consuming method of the three mentioned. The maximum likelihood tree estimation is performed under the assumption of one of five substitution models: the Jukes-Cantor, the Kimura 80, the HKY and the GTR (also known as the REV model) models (see section 30.2.4 for further information about the models). Prior to using the "Maximum Likelihood Phylogeny" tool for creating a phylogenetic tree it is recommended to run the "Model Testing" tool (see section 30.2.3) in order to identify the best suitable models for creating a tree.

- **Create Tree** (⚏) Is a tool that uses distance estimates computed from multiple alignments to create trees. The user can select whether to use Jukes-Cantor distance correction or Kimura distance correction (Kimura 80 for nucleotides/Kimura protein for proteins) in combination with either the neighbor joining or UPGMA method (see section 30.2.5).

30.2.1 K-mer Based Tree Construction

The "K-mer Based Tree Construction" uses single sequences or sequence lists as input and is the simplest way of creating a distance-based phylogenetic tree. To run the "K-mer Based Tree Construction" tool:

Toolbox | Alignments and Trees (📁)| K-mer Based Tree Construction (⚏)

Select sequences or a sequence list (figure 30.1):
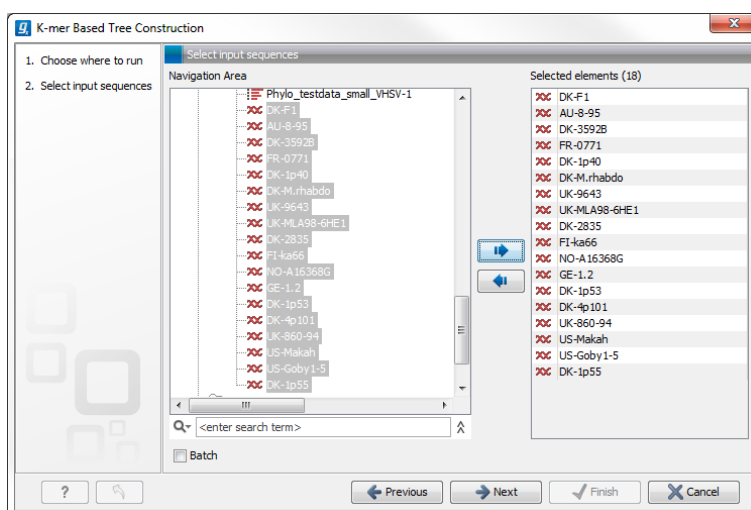
Figure 30.1: *Creating a tree with K-mer based tree construction. Select sequences.*

Next, select the construction method, specify the k-mer length and select a distance measure for tree construction (figure 30.2):

- **Tree construction**

  - **Tree construction method** The user is asked to specify which distance-based method to use for tree construction. There are two options (see section 30.2.5):
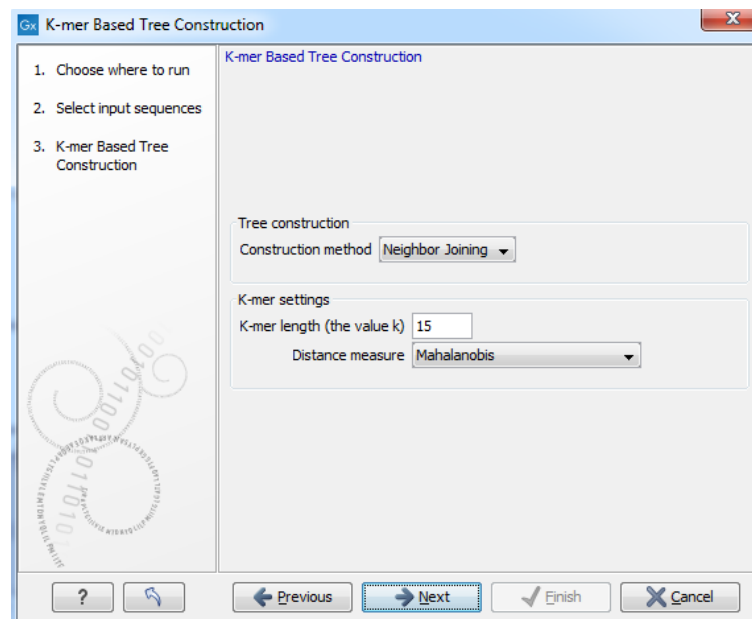
Figure 30.2: *Creating a tree with K-mer based tree construction. Select construction method, specify the k-mer length and select a distance measure.*

　　　　　　　* The **UPGMA** method. Assumes constant rate of evolution.
　　　　　　　* The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.

● **K-mer settings**

　　– **K-mer length (the value k)** Allows specification of the k-mer length, which can be a number between 3 and 50.

　　– **Distance measure** The distance measure is used to compute the distances between two counts of k-mers. Three options exist: Euclidian squared, Mahalanobis, and Fractional common K-mer count. See 30.2.5 for further details.

30.2.2   Create tree

The "Create tree" tool can be used to generate a distance-based phylogenetic tree with multiple alignments as input:

　　　Toolbox | Alignments and Trees (📋 **)| Create Tree (** 🝞 **)**

This will open the dialog displayed in figure 30.3:

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.

Figure 30.4 shows the parameters that can be set for this distance-based tree creation:

● Tree construction (see section 30.2.5)

　　– Tree construction method

　　　　* The **UPGMA** method. Assumes constant rate of evolution.
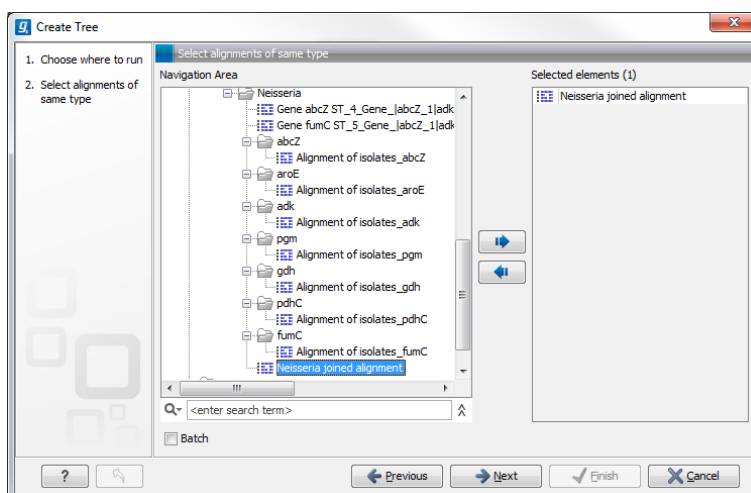
Figure 30.3: *Creating a tree.*
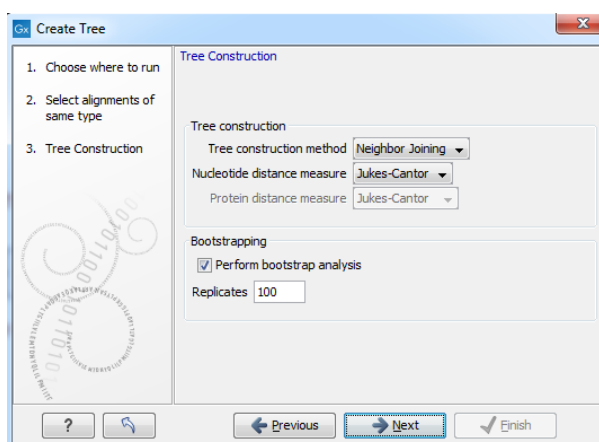
Figure 30.4: *Adjusting parameters for distance-based methods.*

       * The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.

   – Nucleotide distance measure

       * **Jukes-Cantor**. Assumes equal base frequencies and equal substitution rates.

       * **Kimura 80**. Assumes equal base frequencies but distinguishes between transitions and transversions.

   – Protein distance measure

       * **Jukes-Cantor**. Assumes equal amino acid frequency and equal substitution rates.

       * **Kimura protein**. Assumes equal amino acid frequency and equal substitution rates. Includes a small correction term in the distance formula that is intended to give better distance estimates than Jukes-Cantor.

• Bootstrapping.

   – Perform bootstrap analysis. To evaluate the reliability of the inferred trees, *CLC Main Workbench* allows the option of doing a **bootstrap** analysis (see section 30.2.5). A bootstrap value will be attached to each node, and this value is a measure of the confidence in the subtree rooted at the node. The number of replicates used in the bootstrap analysis can be adjusted in the wizard. The default value is 100 replicates which is usually enough to distinguish between reliable and unreliable nodes in the

tree. The bootstrap value assigned to each inner node in the output tree is the percentage (0-100) of replicates which contained the same subtree as the one rooted at the inner node.

For a more detailed explanation, see "Bioinformatics explained" in section 30.2.5.

30.2.3   Model Testing

As the "Model Testing" tool can help identify the best substitution model (30.2.5) to be used for "Maximum Likelihood Phylogeny" tree construction, it is recommended to do "Model Testing" before running the "Maximum Likelihood Phylogeny" tool.

The "Model Testing" tool uses four different statistical analyses:

- Hierarchical likelihood ratio test (hLRT)

- Bayesian information criterion (BIC)

- Minimum theoretical information criterion (AIC)

- Minimum corrected theoretical information criterion (AICc)

to test the substitution models:

- Jukes-Cantor [Jukes and Cantor, 1969]

- Felsenstein 81 [Felsenstein, 1981]

- Kimura 80 [Kimura, 1980]

- HKY [Hasegawa et al., 1985]

- GTR (also known as the REV model) [Yang, 1994a]

To do model testing:

   Toolbox | Alignments and Trees ( )| Model Testing ( )

Select the alignment that you wish to use for the tree construction (figure 30.5):

Specify the parameters to be used for model testing (figure 30.6):

- **Select base tree construction method**

  A base tree (a guiding tree) is required in order to be able to determine which model(s) would be the most appropriate to use to make the best possible phylogenetic tree from a specific alignment. The topology of the base tree is used in the hierarchical likelihood ratio test (hLRT), and the base tree is used as starting point for topology exploration in Bayesian information criterion (BIC), Akaike information criterion (or minimum theoretical information criterion) (AIC), and AICc (AIC with a correction for the sample size) ranking.

    - **Construction method** A base tree is created automatically using one of two methods from the "Create Tree" tool:
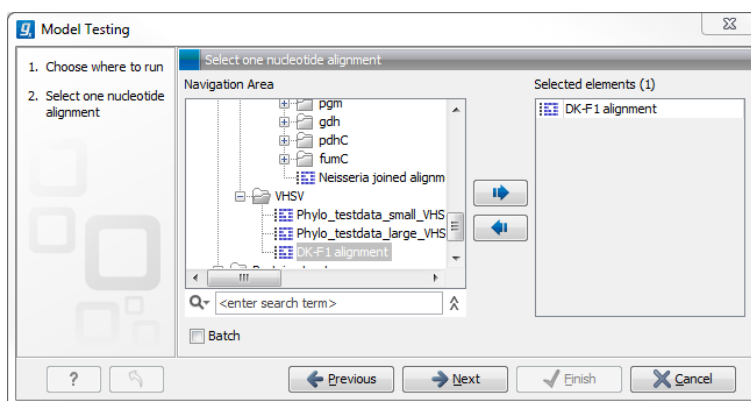
Figure 30.5: *Select alignment for model testing.*
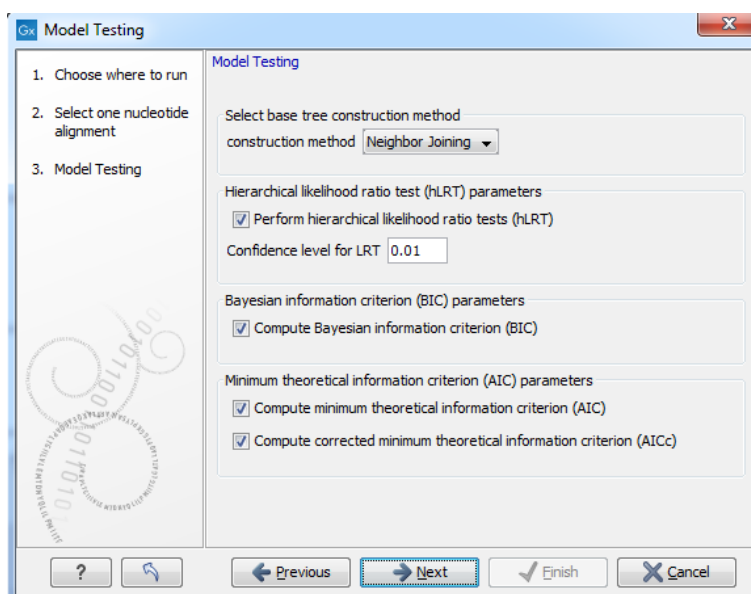
Figure 30.6: *Specify parameters for model testing.*

* The **UPGMA** method. Assumes constant rate of evolution.

* The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.

- **Hierarchical likelihood ratio test (hLRT) parameters** A statistical test of the goodness-of-fit between two models that compares a relatively more complex model to a simpler model to see if it fits a particular dataset significantly better.

  – **Perform hierarchical likelihood ratio test (hLRT)**

  – **Confidence level for LRT** The confidence level used in the likelihood ratio tests.

- **Bayesian information criterion (BIC) parameters**

  – **Compute Bayesian information criterion (BIC)** Rank substitution models based on Bayesian information criterion (BIC). Formula used is BIC = -2ln(L)+Kln(n), where ln(L) is the log-likelihood of the best tree, K is the number of parameters in the model, and ln(n) is the logarithm of the length of the alignment.

- **Minimum theoretical information criterion (AIC) parameters**

– **Compute minimum theoretical information criterion (AIC)** Rank substitution models based on minimum theoretical information criterion (AIC). Formula used is AIC = -2ln(L)+2K, where ln(L) is the log-likelihood of the best tree, K is the number of parameters in the model.

– **Compute corrected minimum theoretical information criterion (AIC)** Rank substitution models based on minimum corrected theoretical information criterion (AICc). Formula used is AICc = -2ln(L)+2K+2K(K+1)/(n-K-1), where ln(L) is the log-likelihood of the best tree, K is the number of parameters in the model, n is the length of the alignment. AICc is recommended over AIC roughly when n/K is less than 40.

The output from model testing is a report that lists all test results in table format. For each tested model the report indicate whether it is recommended to use rate variation or not. Topology variation is recommended in all cases.

From the listed test results, it is up to the user to select the most appropriate model. The different statistical tests will usually agree on which models to recommend although variations may occur. Hence, in order to select the best possible model, it is recommended to select the model that has proven to be the best by most tests.

30.2.4 Maximum Likelihood Phylogeny

To generate a maximum likelihood based phylogenetic tree:

Toolbox | Alignments and Trees (📄)| Maximum Likelihood Phylogeny (🔗)
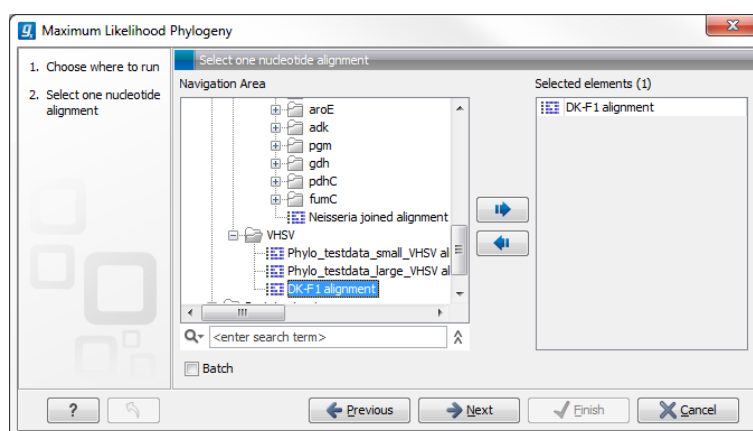
Figure 30.7: *Select the alingment for tree construction*

The following parameters can be set for the maximum likelihood based phylogenetic tree (see figure 30.8):

- **Starting tree**

    – **Construction method** Specify the tree construction method which should be used to create the initial tree. There are two possibilities:
        * Neighbor Joining
        * UPGMA
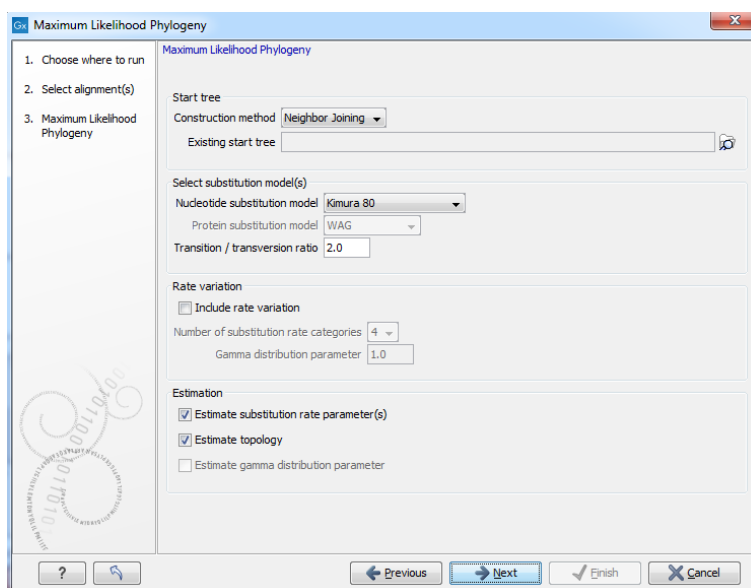
Figure 30.8: *Adjusting parameters for maximum likelihood phylogeny*

- **Exiting start tree** Alternatively, an existing tree can be used as starting tree for the tree reconstruction. Click on the folder icon to the right of the text field to use the browser function to identify the desired starting tree.

• **Select substitution model**

- **Nucleotice substitution model** *CLC Main Workbench* allows maximum likelihood tree estimation to be performed under the assumption of one of five nucleotide substitution models:

  * Jukes-Cantor [Jukes and Cantor, 1969]
  * Felsenstein 81 [Felsenstein, 1981]
  * Kimura 80 [Kimura, 1980]
  * HKY [Hasegawa et al., 1985]
  * General Time Reversible (GTR) (also known as the REV model) [Yang, 1994a]

  All models are time-reversible. In the Kimura 80 and HKY models, the user may set a transtion/transversion ratio value, which will be used as starting value for optimization or as a fixed value, depending on the level of estimation chosen by the user. For further details, see 30.2.5.

- **Protein substitution model** *CLC Main Workbench* allows maximum likelihood tree estimation to be performed under the assumption of one of four protein substitution models:

  * Bishop-Friday [Bishop and Friday, 1985]
  * Dayhoff (PAM) [Dayhoff et al., 1978]
  * JTT [Jones et al., 1992]
  * WAG [Whelan and Goldman, 2001]

The Bishop-Friday substitution model is similar to the Jukes-Cantor model for nucleotide sequences, i.e. it assumes equal amino acid frequencies and substitution rates. This

is an unrealistic assumption and we therefore recommend using one of the remaining three models. The Dayhoff, JTT and WAG substitution models are all based on large scale experiments where amino acid frequencies and substitution rates have been estimated by aligning thousands of protein sequences. For these models, the maximum likelihood tool does not estimate parameters, but simply uses those determined from these experiments.

- **Rate variation**

  To enable variable substitution rates among individual nucleotide sites in the alignment, select 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 discretization 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**.

In the next step of the wizard it is possible to perform bootstrapping (figure 30.9).

- **Bootstrapping**

  - **Perform bootstrap analysis**. To evaluate the reliability of the inferred trees, *CLC Main Workbench* allows the option of doing a **bootstrap** analysis (see section 30.2.5). A bootstrap value will be attached to each node, and this value is a measure of the confidence in the subtree rooted at the node. The number of replicates in the bootstrap analysis can be adjusted in the wizard by specifying the number of times to resample the data. The default value is 100 resamples. The bootstrap value assigned to a node in the output tree is the percentage (0-100) of the bootstrap resamples which resulted in a tree containing the same subtree as that rooted at the node.
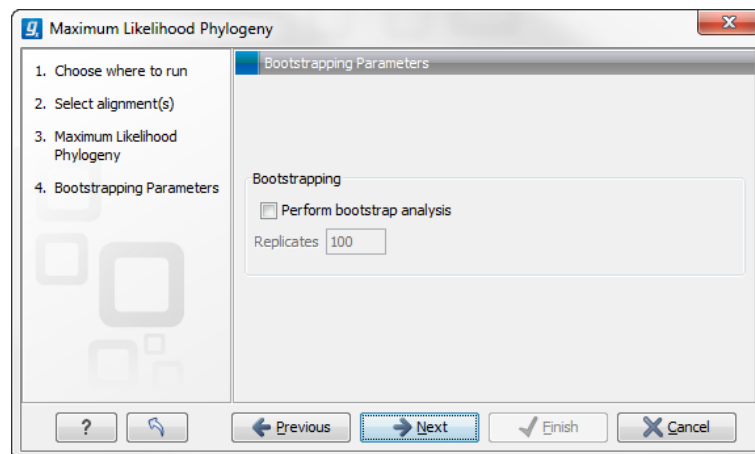
Figure 30.9: *Adjusting parameters for ML phylogeny*

30.2.5 Bioinformatics explained

The phylogenetic tree

The evolutionary hypothesis of a phylogeny can be graphically represented by a phylogenetic tree.

Figure 30.10 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.
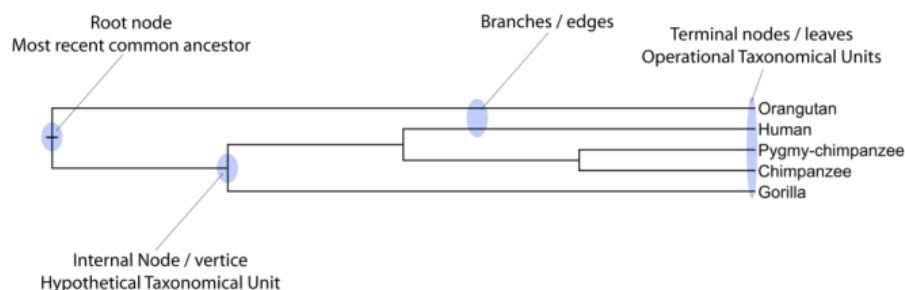
Figure 30.10: *A proposed phylogeny of the great apes (Hominidae). Different components of the tree are marked, see text for description.*

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 30.10 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. In contrast, an unrooted tree would represent relationships without

assumptions about ancestry.

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].

Substitution models and distance estimation

When estimating the evolutionary distance between organisms, one needs a model of how frequently different mutations occur in the DNA. Such models are known as substitution models. Our Model Testing and Maximum Likelihood Phylogeny tools currently support the five nucleotide substitution models listed here:

- Jukes-Cantor [Jukes and Cantor, 1969]

- Felsenstein 81 [Felsenstein, 1981]

- Kimura 80 [Kimura, 1980]

- HKY [Hasegawa et al., 1985]

- GTR (also known as the REV model) [Yang, 1994a]

Common to all these models is that they assume mutations at different sites in the genome occur independently and that the mutations at each site follow the same common probability distribution. Thus all five models provide relative frequencies for each of the 16 possible DNA substitutions (e.g. $C \rightarrow A$, $C \rightarrow C$, $C \rightarrow G$,...).

The Jukes-Cantor and Kimura 80 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 Jukes-Cantor model all substitutions are assumed to occur at equal rates, in the Kimura 80 and HKY models transition and transversion

rates are allowed to differ (substitution between two purines ($A \leftrightarrow G$) or two pyrimidines ($C \leftrightarrow T$) are transitions and purine - pyrimidine substitutions are transversions). The GTR model is the general time reversible model that allows all substitutions to occur at different rates. For the substitution rate matrices describing the substitution models we use the parametrization of Yang [Yang, 1994a].

For protein sequences, our Maximum Likelihood Phylogeny tool supports four substitution models:

- Bishop-Friday [Bishop and Friday, 1985]

- Dayhoff (PAM) [Dayhoff et al., 1978]

- JTT [Jones et al., 1992]

- WAG [Whelan and Goldman, 2001]

As with nucleotide substitution models, it is assumed that mutations at different sites in the genome occur independently and according to the same probability distribution.

The Bishop-Friday model assumes all amino acids occur with same frequency and that all substitutions are equally likely. This is the simplest model, but also the most unrealistic. The remaining three models use amino acid frequencies and substitution rates which have been determined from large scale experiments where huge sets of protein sequences have been aligned and rates have been estimated. These three models reflect the outcome of three different experiments. We recommend using WAG as these rates where estimated from the largest experiment.

K-mer based distance estimation

K-mer based distance estimation is an alternative to estimating evolutionary distance based on multiple alignments. At a high level, the distance between two sequences is defined by first collecting the set of k-mers (subsequences of length k) occuring in the two sequences. From these two sets, the evolutionary distance between the two organisms is now defined by measuring how different the two sets are. The more the two sets look alike, the smaller is the evolutionary distance. The main motivation for estimating evolutionary distance based on k-mers, is that it is computationally much faster than first constructing a multiple alignment. Experiments show that phylogenetic tree reconstruction using k-mer based distances can produce results comparable to the slower multiple alignment based methods [Blaisdell, 1989].

All of the k-mer based distance measures completely ignores the ordering of the k-mers inside the input sequences. Hence, if the selected k value (the length of the sequences) is too small, very distantly related organisms may be assigned a small evolutionary distance (in the extreme case where k is 1, two organisms will be treated as being identical if the frequency of each nucleotide/amino-acid is the same in the two corresponding sequences). In the other extreme, the k-mers should have a length (k) that is somewhat below the average distance between mismatches if the input sequences were aligned (in the extreme case of k=the length of the sequences, two organisms have a maximum distance if they are not identical). Thus the selected k value should not be too large and not too small. A general rule of thumb is to only use k-mer based distance estimation for organisms that are not too distantly related.

Formal definition of distance. In the following, we give a more formal definition of the three supported distance measures: Euclidian-squared, Mahalanobis and Fractional common k-mer

count. For all three, we first associate a point $p(s)$ to every input sequence s. Each point $p(s)$ has one coordinate for every possible length k sequence (e.g. if s represent nucleotide sequences, then $p(s)$ has 4^k coordinates). The coordinate corresponding to a length k sequence x has the value: ''number of times x occurs as a subsequence in s''. Now for two sequences s_1 and s_2, their evolutionary distance is defined as follows:

- **Euclidian squared**: For this measure, the distance is simply defined as the (squared Euclidian) distance between the two points $p(s_1)$ and $p(s_2)$, i.e.

$$\text{dist}(s_1, s_2) = \sum_i (p(s_1)_i - p(s_2)_i)^2.$$

- **Mahalanobis**: This measure is essentially a fine-tuned version of the Euclidian squared distance measure. Here all the counts $p(s_j)_i$ are ''normalized'' by dividing with the standard deviation σ_j of the count for the k-mer. The revised formula thus becomes:

$$\text{dist}(s_1, s_2) = \sum_i (p(s_1)_i/\sigma_i - p(s_2)_i/\sigma_i)^2.$$

Here the standard deviations can be computed directly from a set of equilibrium frequencies for the different bases, see [Gentleman and Mullin, 1989].

- **Fractional common k-mer count**: For the last measure, the distance is computed based on the minimum count of every k-mer in the two sequences, thus if two sequences are very different, the minimums will all be small. The formula is as follows:

$$\text{dist}(s_1, s_2) = \log(0.1 + \sum_i (\min(p(s_1)_i, p(s_2)_i)/(\min(n, m) - k + 1))).$$

Here n is the length of s_1 and m is the length of s_2. This method has been described in [Edgar, 2004].

In experiments performed in [Höhl et al., 2007], the Mahalanobis distance measure seemed to be the best performing of the three supported measures.

Distance based reconstruction methods

Distance based phylogenetic reconstruction methods use a pairwise distance estimate between the input organisms to reconstruct trees. The distances are an estimate of the evolutionary distance between each pair of organisms which are usually computed from DNA or amino acid sequences. Given two homologous sequences a distance estimate can be computed by aligning the sequences and then counting the number of positions where the sequences differ. The number of differences is called the observed number of substitutions and is usually an underestimate of the real distance as multiple mutations could have occurred at any position. To correct for these hidden substitutions a substitution model, such as Jukes-Cantor or Kimura 80, can be used to get a more precise distance estimate (see section 30.2.5). Alternatively, k-mer based methods or SNP based methods can be used to get a distance estimate without the use of substitution models.

After distance estimates have been computed, a phylogenetic tree can be reconstructed using a distance based reconstruction method. Most distance based methods perform a bottom up

reconstruction using a greedy clustering algorithm. Initially, each input organism is put in its own cluster which corresponds to a leaf node in the resulting tree. Next, pairs of clusters are iteratively joined into higher level clusters, which correspond to connecting two nodes in the tree with a new parent node. When a single node remains, the tree is reconstructed.

The *CLC Main Workbench* provides two of the most widely used distance based reconstruction methods:

- The **UPGMA** method [Michener and Sokal, 1957] which assumes a constant rate of evolution (molecular clock hypothesis) in the different lineages. This method reconstruct trees by iteratively joining the two nearest clusters until there is only one cluster left. The result of the UPGMA method is a rooted bifurcating tree annotated with branch lengths.

- The **Neighbor Joining** method [Saitou and Nei, 1987] attempts to reconstruct a minimum evolution tree (a tree where the sum of all branch lengths is minimized). Opposite to the UPGMA method, the neighbour joining method is well suited for trees with varying rates of evolution in different lineages. A tree is reconstructed by iteratively joining clusters which are close to each other but at the same time far from all other clusters. The resulting tree is a bifurcating tree with branch lenghts. Since no particular biological hypothesis is made about the placement of the root in this method, the resulting tree is unrooted.

Maximum likelihood reconstruction methods

Maximum likelihood (ML) based reconstruction methods [Felsenstein, 1981] seek to identify the most probable tree given the data available, i.e. maximize $P(tree|data)$ where the $tree$ refers to a tree topology with branch lengths while $data$ is usually a set of sequences. However, it is not possible to compute $P(tree|data)$ so instead ML based methods have to compute the probability of the data given a tree, i.e. $P(data|tree)$. The ML tree is then the tree which makes the data most probable. In other words, ML methods search for the tree that gives the highest probability of producing the observed sequences. This is done by searching through the space of all possible trees while computing an ML estimate for each tree. Computing an ML estimate for a tree is time consuming and since the number of tree topologies grows exponentially with the number of leaves in a tree, it is infeasible to explore all possible topologies. Consequently, ML methods must employ search heuristics that quickly converges towards a tree with a likelihood close to the real ML tree.

The likelihood of trees are computed using an explicit model of evolution such as the Jukes-Cantor or Kimura 80 models. Choosing the right model is often important to get a good result and to help users choose the correct model for a data, set the "Model Testing" tool (see section 30.2.3) can be used to test a range of different models for nucleotide input sequences.

The search heuristics which are commonly used in ML methods requires an initial phylogenetic tree as a starting point for the search. An initial tree which is close to the optimal solution, can reduce the running time of ML methods and improve the chance of finding a tree with a large likelihood. A common way of reconstructing a good initial tree is to use a distance based method such as UPGMA or neighbour-joining to produce a tree based on a multiple alignment.

Bootstrap tests

Bootstrap tests [Felsenstein, 1985] is one of the most common ways to evaluate the reliability of the topology of a phylogenetic tree. In a bootstrap test, trees are evaluated using Efron's re-sampling technique [Efron, 1982], which samples nucleotides from the original set of sequences as follows:

Given an alignment of n sequences (rows) of length l (columns), we randomly choose l columns in the alignment with replacement and use them to create a new alignment. The new alignment has n rows and l columns just like the original alignment but it may contain duplicate columns and some columns in the original alignment may not be included in the new alignment. From this new alignment we reconstruct the corresponding tree and compare it to the original tree. For each subtree in the original tree we search for the same subtree in the new tree and add a score of one to the node at the root of the subtree if the subtree is present in the new tree. This procedure is repeated a number of times (usually around 100 times). The result is a counter for each interior node of the original tree, which indicate how likely it is to observe the exact same subtree when the input sequences are sampled. A bootstrap value is then computed for each interior node as the percentage of resampled trees that contained the same subtree as that rooted at the node.

Bootstrap values can be seen as a measure of how reliably we can reconstruct a tree, given the sequence data available. If all trees reconstructed from resampled sequence data have very different topologies, then most bootstrap values will be low, which is a strong indication that the topology of the original tree cannot be trusted.

Scale bar

The scale bar unit depends on the distance measure used and the tree construction algorithm used. The trees produced using the Maximum Likelihood Phylogeny tool has a very specific interpretation: A distance of x means that the expected number of substitutions/changes per nucleotide (amino acid for protein sequences) is x. i.e. if the distance between two taxa is 0.01, you expected a change in each nucleotide independently with probability 1 %. For the remaining algorithms, there is not as nice an interpretation. The distance depends on the weight given to different mutations as specified by the distance measure.

30.3 Tree Settings

The Tree Settings Side Panel found in the left side of the view area can be used to adjust the tree layout and to visualize metadata that is associated with the tree nodes.

The preferred tree layout settings (user defined tree settings) can be saved and applied via the top right **Save Tree Settings** (figure 30.11). Settings can either be saved **For This Tree Only** or for all saved phylogenetic trees (**For Tree View in General**). The first option will save the layout of the tree for that tree only and it ensures that the layout is preserved even if it is exported and opened by a different user. The second option stores the layout globally in the Workbench and makes it available to other trees through the **Apply Saved Settings** option.

Tree Settings contains the following categories:

- Minimap
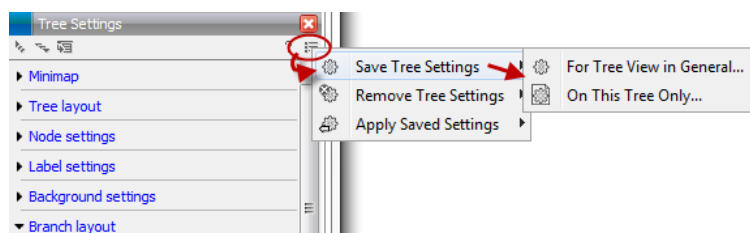
Figure 30.11: *Save, remove or apply preferred layout settings.*

- Tree layout

- Node settings

- Label settings

- Background settings

- Branch layout

- Bootstrap settings

- Metadata

30.3.1 Minimap

The Minimap is a navigation tool that shows a small version of the tree. A grey square indicates the specific part of the tree that is visible in the View Area (figure 30.12). To navigate the tree using the Minimap, click on the Minimap with the mouse and move the grey square around within the Minimap.
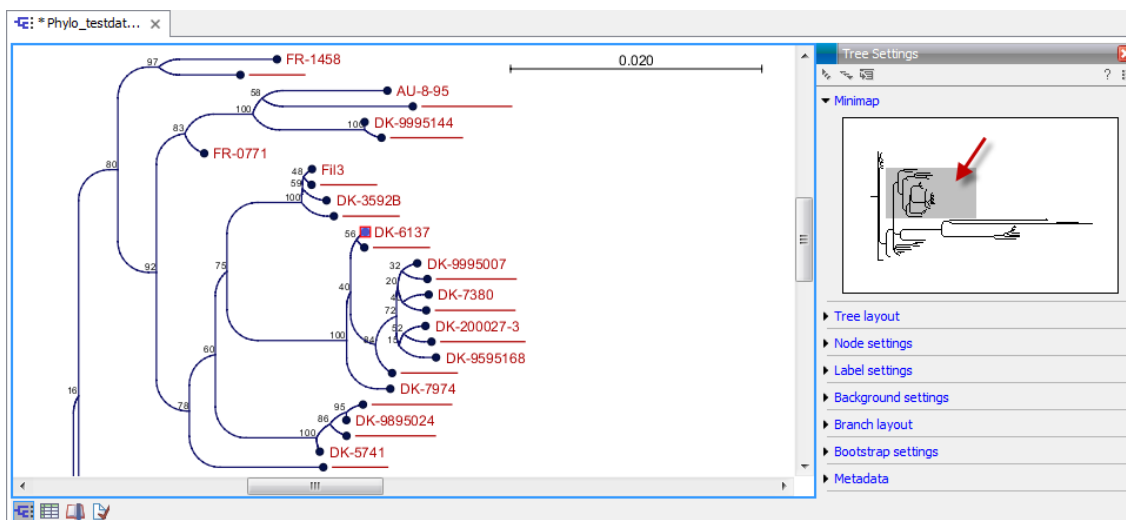
Figure 30.12: *Visualization of a phylogenetic tree. The grey square in the Minimap shows the part of the tree that is shown in the View Area.*

30.3.2 Tree layout

The **Tree Layout** can be adjusted in the Side Panel (figure 30.13).
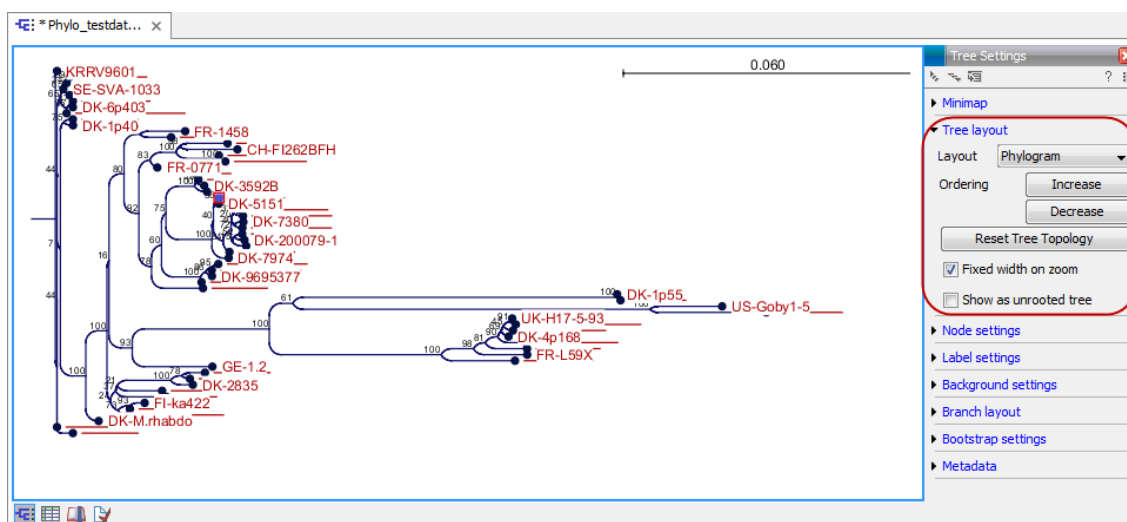
Figure 30.13: *The tree layout can be adjusted in the Side Panel. Five different layouts can be selected and the node order can be changed to increasing or decreasing. The tree topology and node order can be reverted to the original view with the button labeled "Reset Tree Topology".*

- **Layout** Selects the overall outline of the five layout types: Phylogram, Cladogram, Circular Phylogram, Circular Cladogram or Radial.

  - **Phylogram** is a rooted tree where the edges have "lengths", usually proportional to the inferred amount of evolutionary change to have occurred along each branch.
  - **Cladogram** is a rooted tree without branch lengths which is useful for visualizing the topology of trees.
  - **Circular Phylogram** is also a phylogram but with the leaves in a circular layout.
  - **Circular Cladogram** is also a cladogram but with the leaves in a circular layout.
  - **Radial** is an unrooted tree that has the same topology and branch lengths as the rooted styles, but lacks any indication of evolutionary direction.

- **Ordering** The nodes can be ordered after the branch length; either **Increasing** (shown in figure 30.14) or **Decreasing**.

- **Reset Tree Topology** Resets to the default tree topology and node order (see figure 30.14).

- **Ordering** The nodes can be ordered after the branch length; either **Increasing** (shown in figure 30.14) or **Decreasing**.

- **Reset Tree Topology** Resets to the default tree topology and node order (see figure 30.14).

- **Fixed width on zoom** Locks the horizontal size of the tree to the size of the main window. Zoom is therefore only performed on the vertical axis when this option is enabled.

- **Show as unrooted tree** The tree can be shown with or without a root.

30.3.3 Node settings

The nodes can be manipulated in several ways.

Figure 30.14: *The tree layout can be adjusted in the Side Panel. The top part of the figure shows a tree with increasing node order. In the bottom part of the figure the tree has been reverted to the original tree topology.*

- **Leaf node symbol** Leaf nodes can be shown as a range of different symbols (Dot, Box, Circle, etc.).

- **Internal node symbols** The internal nodes can also be shown with a range of different symbols (Dot, Box, Circle, etc.).

- **Max. symbol size** The size of leaf- and internal node symbols can be adjusted.

- **Avoid overlapping symbols** The symbol size will be automatically limited to avoid overlaps between symbols in the current view.

- **Node color** Specify a fixed color for all nodes in the tree.

The node layout settings in the Side Panel are shown in figure 30.15.

30.3.4   Label settings

- **Label font settings** Can be used to specify/adjust font type, size and typography (Bold, Italic or normal).
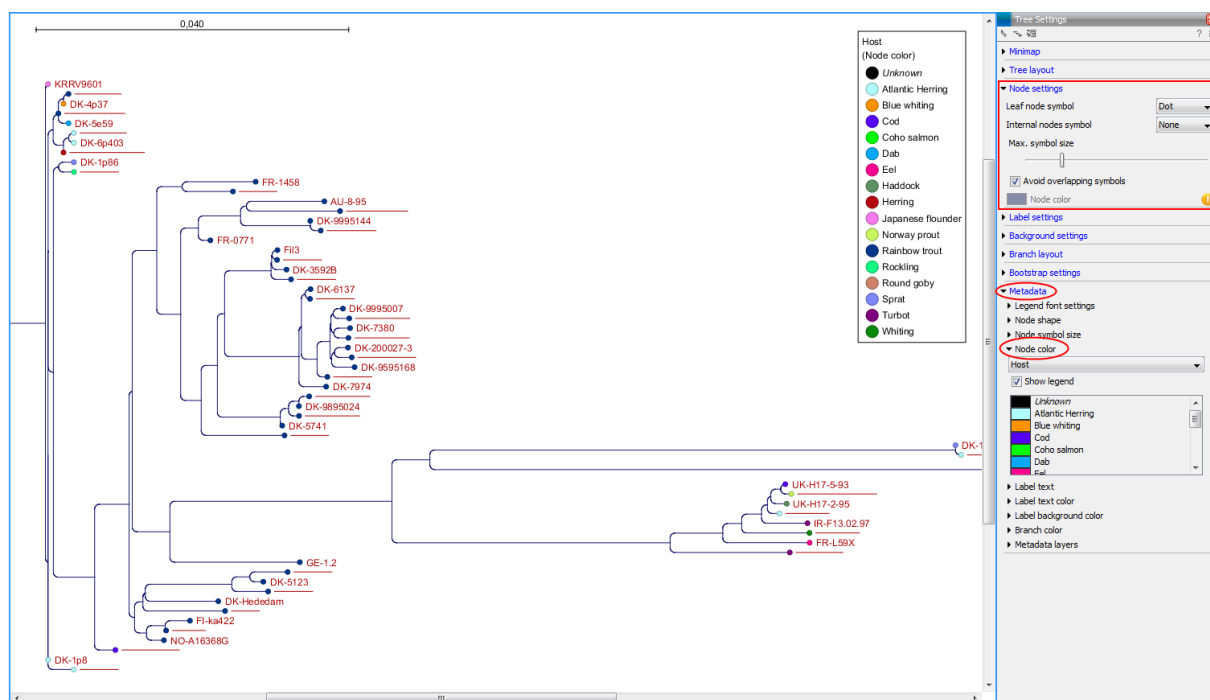
Figure 30.15: *The Node Layout settings. Node color is specified by metadata and is therefore inactive in this example.*

- **Hide overlapping labels** Disable automatic hiding of overlapping labels and display all labels even if they overlap.

- **Show internal node labels** Labels for internal nodes of the tree (if any) can be displayed. Please note that subtrees and nodes can be labeled with a custom text. This is done by right clicking the node and selecting **Edit Label** (see figure 30.16).

- **Show leaf node labels** Leaf node labels can be shown or hidden.

- **Rotate Subtree labels** Subtree labels can be shown horizontally or vertically. Labels are shown vertically when "Rotate subtree labels" has been selected. Subtree labels can be added with the right click option "Set Subtree Label" that is enabled from "Decorate subtree" (see section 30.3.9).

- **Align labels** Align labels to the node furthest from the center of the tree so that all labels are positioned next to each other. The exact behavior depends on the selected tree layout.

- **Connect labels to nodes** Adds a thin line from the leaf node to the aligned label. Only possible when Align labels option is selected.

When working with big trees there is typically not enough space to show all labels. As illustrated in figure 30.16, only some of the labels are shown. The hidden labels are illustrated with thin horizontal lines (figure 30.17).

There are different ways of showing more labels. One way is to reduce the font size of the labels, which can be done under **Label font settings** in the Side Panel. Another option is to zoom in on specific areas of the tree (figure 30.17 and figure 30.18). The last option is to disable **Hide overlapping labels** under "Label settings" in the right side panel. When this option is unchecked
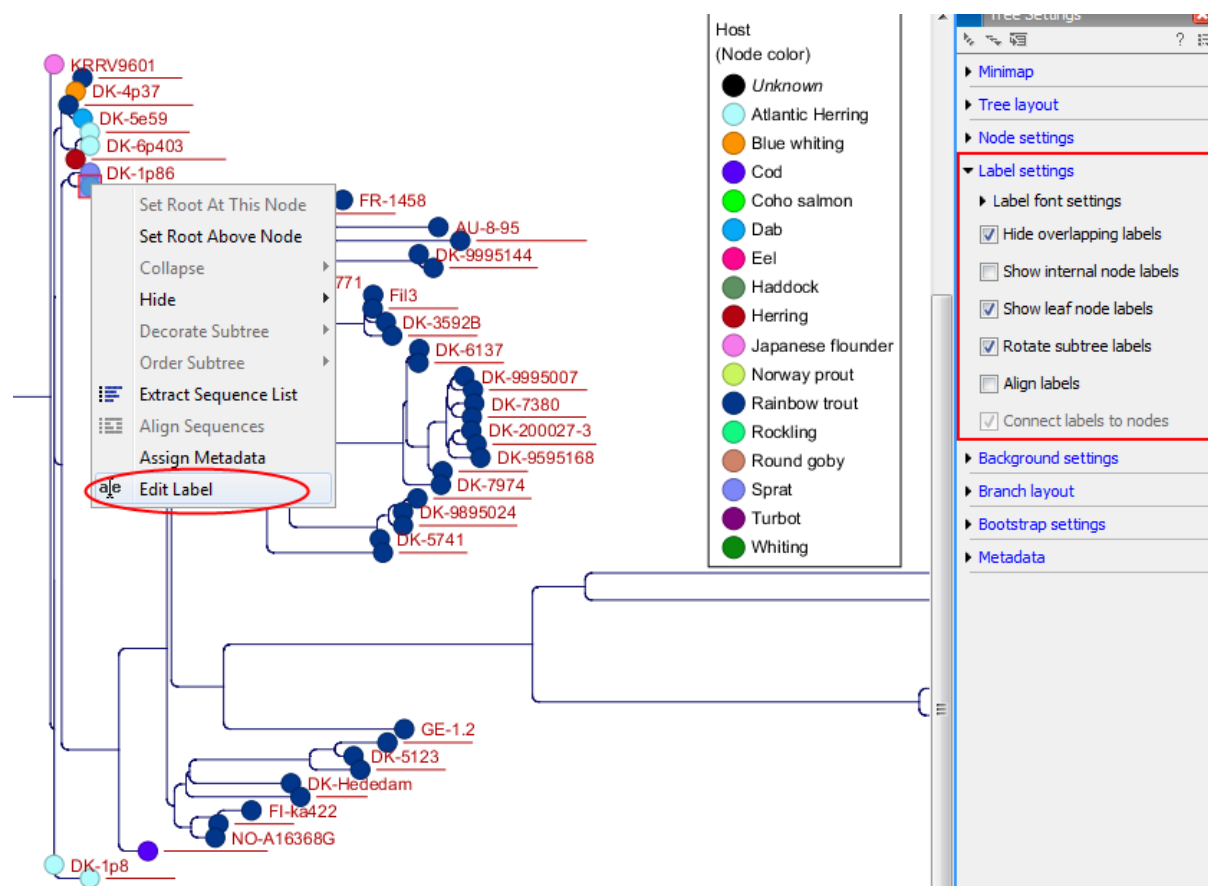
Figure 30.16: *"Edit label" in the right click menu can be used to customize the label text. The way node labels are displayed can be controlled through the labels settings in the right side panel.*

all labels are shown even if the text overlaps. When allowing overlapping labels it is usually a good idea to disable **Show label background** under "Background settings" (see section 30.3.5).

Note! When working with a tree with hidden labels, it is possible to make the hidden label text appear by moving the mouse over the node with the hidden label.

30.3.5   Background settings

- **Show label background** Show a background color for each label. Once ticked, it is possible to specify whether to use a fixed color or to use the color that is associated with the selected metadata category.

30.3.6   Branch layout

- **Branch length font settings** Specify/adjust font type, size and typography (Bold, Italic or normal).

- **Line color** Select the default line color.

- **Line width** Select the width of branches (1.0-3.0 pixels).

- **Curvature** Adjust the degree of branch curvature to get branches with round corners.
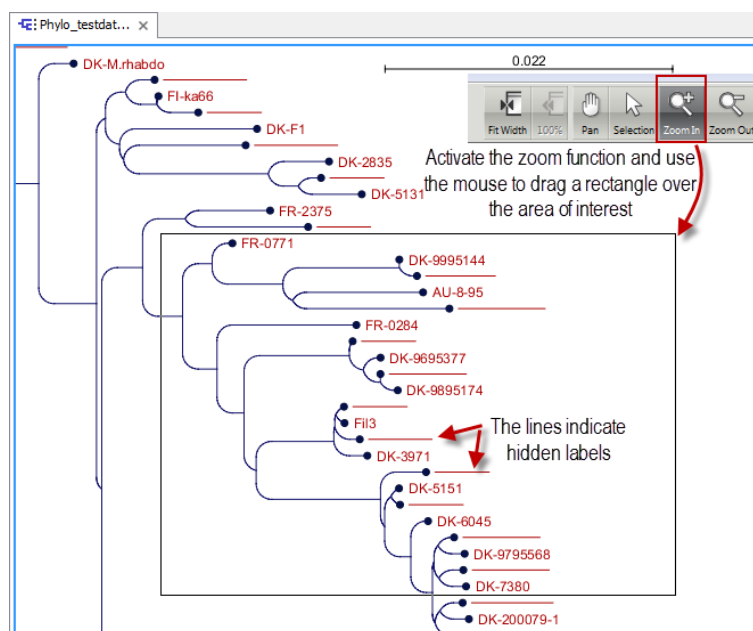
Figure 30.17: *The zoom function in the upper right corner of CLC Genomics Workbench can be used to zoom in on a particular region of the tree. When the zoom function has been activated, use the mouse to drag a rectangle over the area that you wish to zoom in at.*
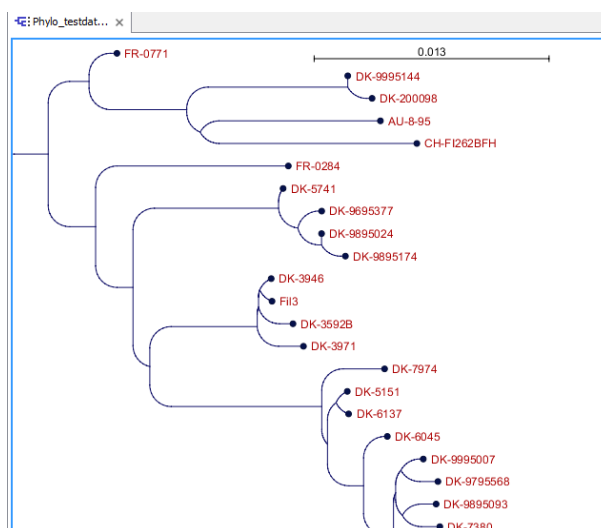
Figure 30.18: *After zooming in on a region of interest more labels become visible. In this example all labels are now visible.*

- **Min. length** Select a minimum branch length. This option can be used to prevent nodes connected with a short branch to cluster at the parent node.

- **Show branch lengths** Show or hide the branch lengths.

The branch layout settings in the Side Panel are shown in figure 30.19.

30.3.7 Bootstrap settings

Bootstrap values can be shown on the internal nodes. The bootstrap values are shown in percent and can be interpreted as confidence levels where a bootstrap value close to 100 indicate a
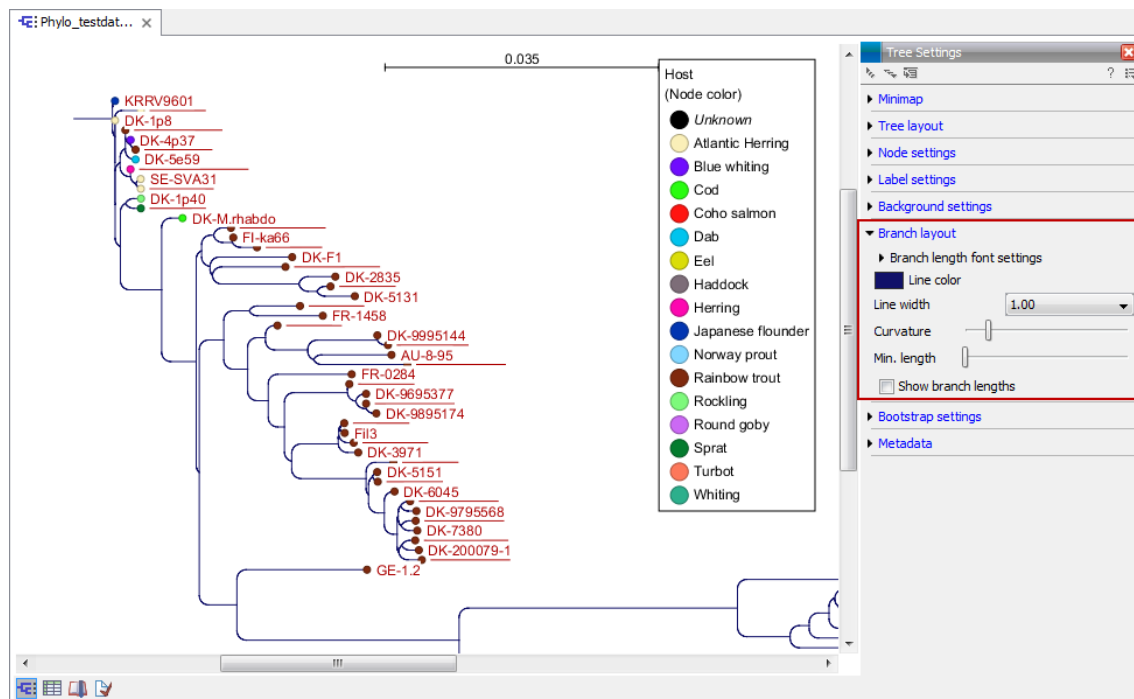
Figure 30.19: *Branch Layout settings.*

clade, which is strongly supported by the data from which the tree was reconstructed. Bootstrap values are useful for identifying clades in the tree where the topology (and branch lengths) should not be trusted.

- **Bootstrap value font settings** Specify/adjust font type, size and typography (Bold, Italic or normal).

- **Show bootstrap values (%)** Show or hide bootstrap values. When selected, the bootstrap values (in percent) will be displayed on internal nodes if these have been computed during the reconstruction of the tree.

- **Bootstrap threshold (%)** When specifying a bootstrap threshold, the branch lengths can be controlled manually by collapsing internal nodes with bootstrap values under a certain threshold.

- **Highlight bootstrap $\geq$ (%)** Highlights branches where the bootstrap value is above the user defined threshold.

30.3.8   Metadata

Metadata associated with a phylogenetic tree (described in detail in section 30.4) can be visualized in a number of different ways:

- **Node shape** Different node shapes are available to visualize metadata.

- **Node symbol size** Change the node symbol size to visualize metadata.

- **Node color** Change the node color to visualize metadata.

- **Label text** The metadata can be shown directly as text labels as shown in figure 30.20.

- **Label text color** The label text can be colored and used to visualize metadata (see figure 30.20).

- **Label background color** The background color of node text labels can be used to visualize metadata.

- **Branch color** Branch colors can be changed according to metadata.

- **Metadata layers** Color coded layers shown next to leaf nodes.

Please note that when visualizing metadata through a tree property that can be adjusted in the right side panel (such as node color or node size), an exclamation mark will appear next to the control for that property to indicate that the setting is inactive because it is defined by metadata (see figure 30.15).
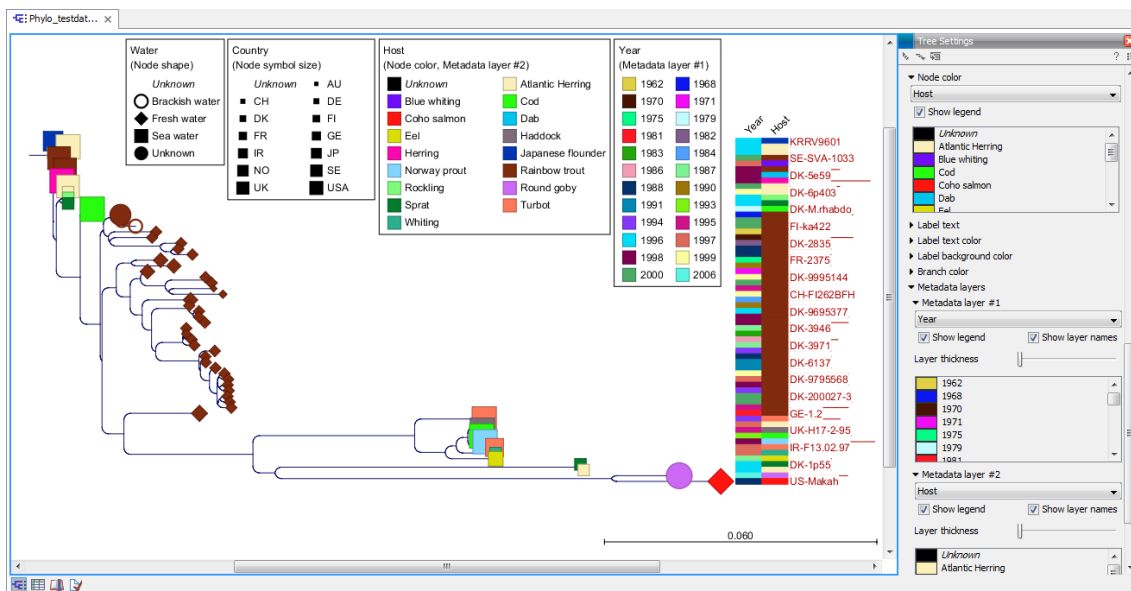
Figure 30.20: *Different types of metadata kan be visualized by adjusting node size, shape, and color. Two color-code metadata layers (Year and Host) are shown in the right side of the tree.*

30.3.9 Node right click menu

Additional options for layout and extraction of subtree data are available when right clicking the nodes (figure 30.16):

- **Set Root At This Node** Re-root the tree using the selected node as root. Please note that re-rooting will change the tree topology.

- **Set Root Above Node** Re-root the tree by inserting a node between the selected node and its parent. Useful for rooting trees using an outgroup.

- **Collapse** Branches associated with a selected node can be collapsed with or without the associated labels. Collapsed branches can be uncollapsed using the *Uncollapse* option in the same menu.
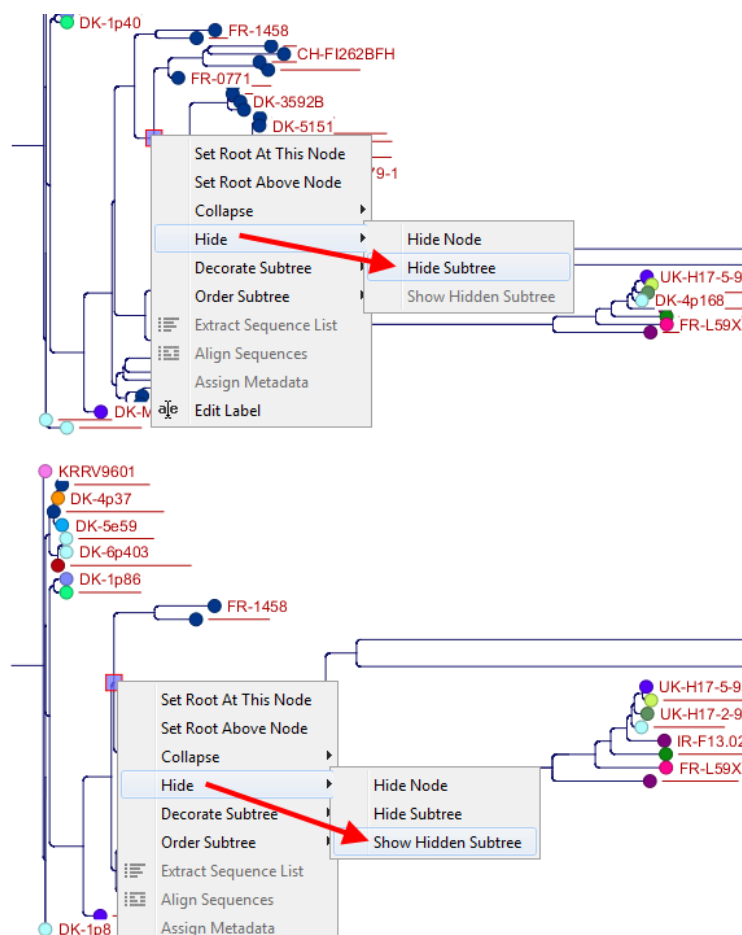
Figure 30.21: *A subtree can be hidden by selecting "Hide Subtree" and is shown again when selecting "Show Hidden Subtree" on a parent node.*
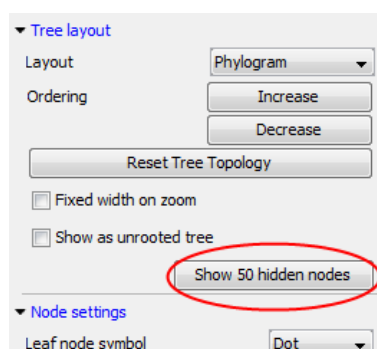
Figure 30.22: *When hiding nodes, a new button labeled "Show X hidden nodes" appears in the Side Panel under "Tree Layout". When pressing this button, all hidden nodes are brought back.*

- **Hide** Can be used to hide a node or a subtree. Hidden nodes or subtrees can be shown again using the *Show Hidden Subtree* function on a node which is root in a subtree containing hidden nodes (see figure 30.21). When hiding nodes, a new button appears labeled "Show X hidden nodes" in the Side Panel under "Tree Layout" (figure 30.22). When pressing this button, all hidden nodes are shown again.

- **Decorate Subtree** A subtree can be labeled with a customized name, and the subtree lines and/or background can be colored.

- **Order Subtree** Rearrange leaves and branches in a subtree by Increasing/Decreasing depth, respectively. Alternatively, change the order of a node's children by left clicking and dragging one of the node's children.

- **Extract Sequence List** Sequences associated with selected leaf nodes are extracted to a new sequence list.

- **Align Sequences** Sequences associated with selected leaf nodes are extracted and used as input to the *Create Alignment* tool.

- **Assign Metadata** Metadata can be added, deleted or modified. To add new metadata categories a new "Name" must be assigned. (This will be the column header in the metadata table). To add a new metadata category, enter a value in the "Value" field. To delete values, highlight the relevant nodes and right click on the selected nodes. In the dialog that appears, use the drop-down list to select the name of the desired metadata category and leave the value field empty. When pressing "Add" the values for the selected metadata category will be deleted from the selected nodes. Metadata can be modified in the same way, but instead of leaving the value field empty, the new value should be entered.

- **Edit label** Edit the text in the selected node label. Labels can be shown or hidden by using the Side Panel: **Label settings | Show internal node labels**

30.4   Metadata and Phylogenetic Trees

When a tree is reconstructed, some mandatory metadata will be added to nodes in the tree. These metadata are special in the sense that the tree viewer has specialized features for visualizing the data and some of them cannot be edited. The mandatory metadata include:

- **Node name** The node name.

- **Branch length** The length of the branch, which connects a node to the parent node.

- **Bootstrap value** The bootstrap value for internal nodes.

- **Size** The length of the sequence which corresponds to each leaf node. This only applies to leaf nodes.

- **Start of sequence** The first 50bp of the sequence corresponding to each leaf node.

To view metadata associated with a phylogenetic tree, click on the table icon (▦) at the bottom of the tree. If you hold down the Ctrl key (or (⌘) on Mac) while clicking on the table icon (▦), you will be able to see both the tree and the table in a split view (figure 30.23).

Additional metadata can be associated with a tree by clicking the **Import Metadata** button. This will open up the dialog shown in figure 30.24.

To associate metadata with an existing tree a common denominator is required. This is achieved by mapping the node names in the "Name" column of the metadata table to the names that have been used in the metadata table to be imported. In this example the "Strain" column holds
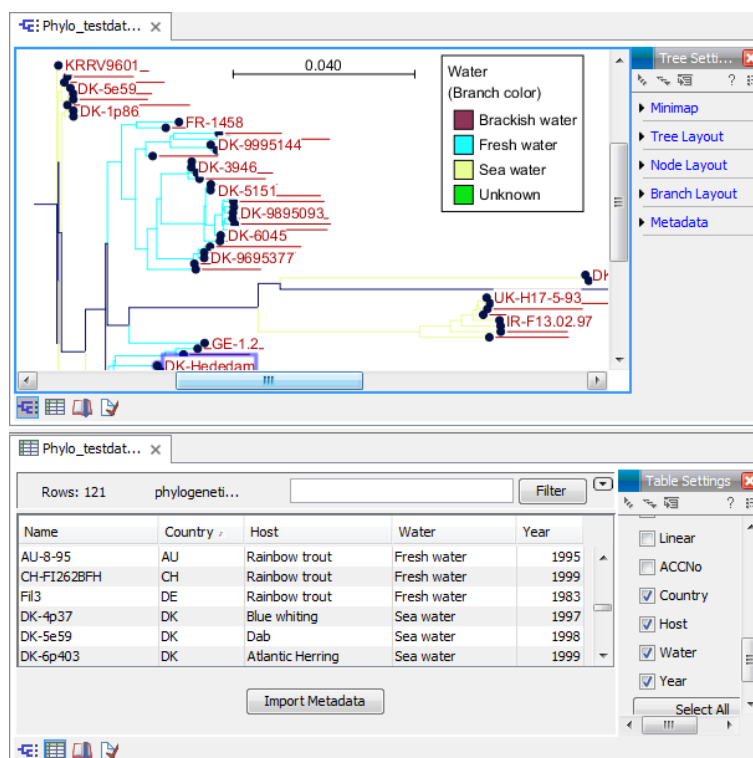
Figure 30.23: *Tabular metadata that is associated with an existing tree shown in a split view.*

the names of the nodes and this column must be assigned "Name" to allow the importer to associate metadata with nodes in the tree.

It is possible to import a subset of the columns in a set of metadata. An example is given in figure 30.24. The column "H" is not relevant to import and can be excluded simply by leaving the text field at the top row of the column empty.

30.4.1 Table Settings and Filtering

How to use the metadata table (see figure 30.25):

- **Column width** The column width can be adjusted in two ways; *Manually* or *Automatically*.

- **Show column** Selects which metadata categories that are shown in the table layout.

- **Filtering Metadata information** Metadata information in a table can be filtered by a simple- or advanced mode (this is described in the CLC Genomics Workbench manual, Appendix D, Filtering tables).

30.4.2 Add or modify metadata

It is possible to add and modify metadata from both the tree view and the table view.

Metadata can be added and edited in the metadata table by using the following right click options (see figure 30.26):

- **Assign Metadata** The right click option "Assign Metadata" can be used for four purposes.
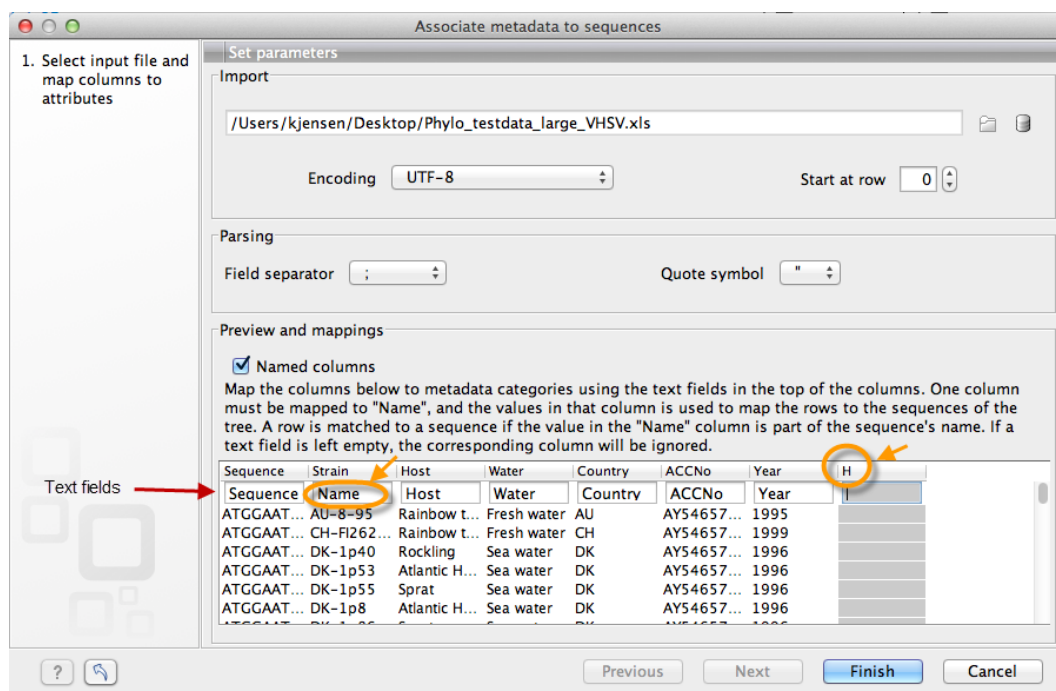
Figure 30.24: *Import of metadata for a tree. The second column named "Strain" is choosen as the common denominator by entering "Name" in the text field of the column. The column labeled "H" is ignored by not assigning a column heading to this column.*

- – To add new metadata categories (columns). In this case, a new "Name" must be assigned, which will be the column header. To add a new column requires that a value is entered in the "Value" field. This can be done by right clicking anywhere in the table.

- – To add values to one or more rows in an existing column. In this case, highlight the relevant rows and right click on the selected rows. In the dialog that appears, use the drop-down list to select the name of the desired column and enter a value.

- – To delete values from an existing column. This is done in the same way as when adding a new value, with the only exception that the value field should be left empty.

- – To delete metadata columns. This is done by selecting all rows in the table followed by a right click anywhere in the table. Select the name of the column to delete from the drop down menu and leave the value field blank. When pressing "Add", the selected column will disappear.

- • **Delete Metadata "column header"** This is the most simple way of deleting a metadata column. Click on one of the rows in the column to delete and select "Delete *column header*".

- • **Edit "column header"** To modify existing metadata point, right click on a cell in the table and select the "Edit *column header*" (see an example in figure 30.27). To edit multiple entries at once, select multiple rows in the table, right click a selected cell in the column you want to edit and choose "Edit *column header*". This will change values in all selected rows in the column that was clicked.

Figure 30.25: *Metadata table. The column width can be adjusted manually or automatically. Under "Show column" it is possible to select which columns should be shown in the table. Filtering using specific criteria can be performed (this is described in the CLC Genomics Workbench manual, Appendix D, Filtering tables).*

Figure 30.26: *Right click options in the metadata table.*

30.4.3 Unknown metadata values

When visualizing a metadata category where one or more nodes in the tree have undefined values, these nodes will be visualized using a *default* value. This value will always be shown in italics in the top of the legend (see figure 30.28). To remove this entry in the legend, all nodes must have a value assigned in the corresponding metadata category.

30.4.4 Selection of specific nodes

Selection of nodes in a tree is automatically synchronized to the metadata table and the other way around. Nodes in a tree can be selected in three ways:

- *Selection of a single node* Click once on a single node. Additional nodes can be added by holding down Ctrl (or (⌘) for Mac) and clicking on them (see figure 30.29).

- *Selecting all nodes in a subtree* Double clicking on a inner node results in the selection of all nodes in the subtree rooted at the node.
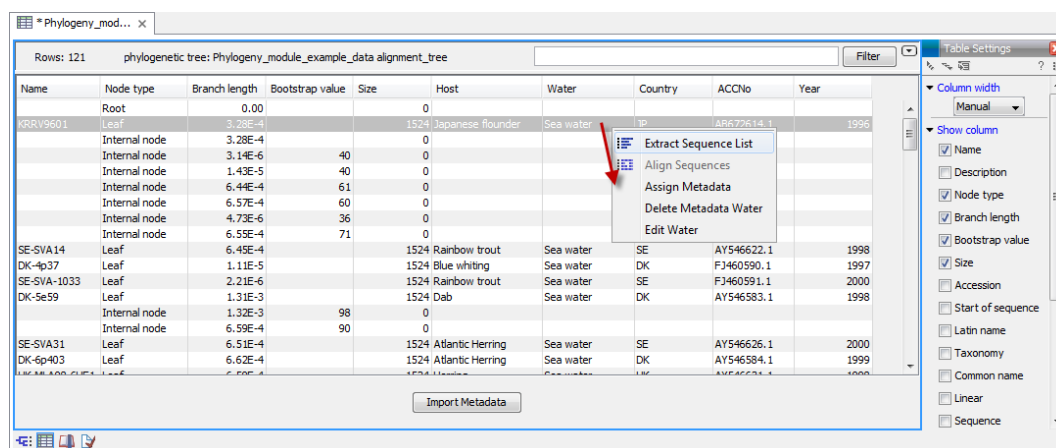
Figure 30.27: *To include an extra metadata column, use the right click option "Assign Metadata", provide "Name" (the column header) and "Value". To modify existing metadata, click on the specific field, select "Edit column header" and provide new value.*
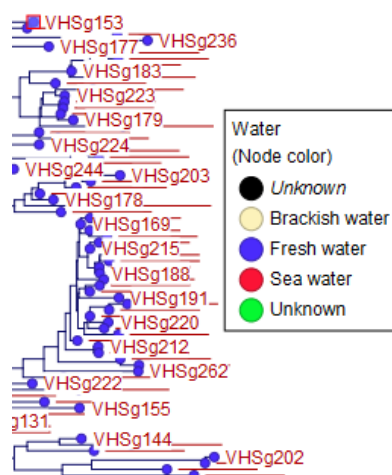
Figure 30.28: *A legend for a metadata category where one or more values are undefined.*

- *Selection via the Metadata table* Select one or more entries in the table. The corresponding nodes will now be selected in the tree.

It is possible to extract a subset of the underlying sequence data directly through either the tree viewer or the metadata table as follows. Select one or more nodes in the tree where at least one node has a sequence attached. Right click one of the selected nodes and choose **Extract Sequence List**. This will generate a new sequence list containing all sequences attached to the selected nodes. The same functionality is available in the metadata table where sequences can be extracted from selected rows using the right click menu. Please note that all extracted sequences are copies and any changes to these sequences will not be reflected in the tree.

When analyzing a phylogenetic tree it is often convenient to have a multiple alignment of sequences from e.g. a specific clade in the tree. A quick way to generate such an alignment is to first select one or more nodes in the tree (or the corresponding entries in the metadata table) and then select **Align Sequences** in the right click menu. This will extract the sequences corresponding to the selected elements and use a copy of them as input to the multiple alignment tool (see section 29.6). Next, change relevant option in the multiple alignment wizard that pops

up and click **Finish**. The multiple alignment will now be generated.
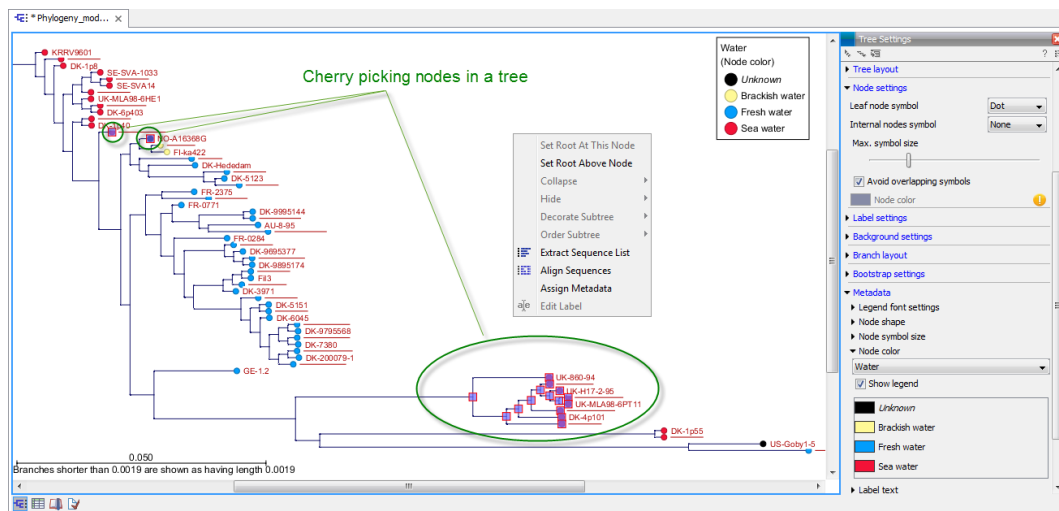
Figure 30.29: *Cherry picking nodes in a tree. The selected leaf sequences can be extracted by right clicking on one of the selected nodes and selecting "Extract Sequence List". It is also possible to Align Sequences directly by right clicking on the nodes or leaves.*

Chapter 31

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

624

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 Main 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

31.1   RNA secondary structure prediction

CLC Main 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 31.5.

In *CLC Main 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/`.

31.1.1   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 31.1.

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
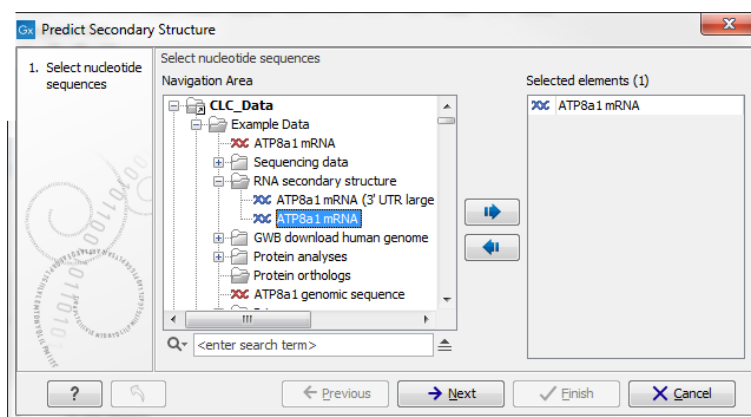
Figure 31.1: *Selecting RNA or DNA sequences for structure prediction (DNA is folded as if it was RNA).*

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 31.2.
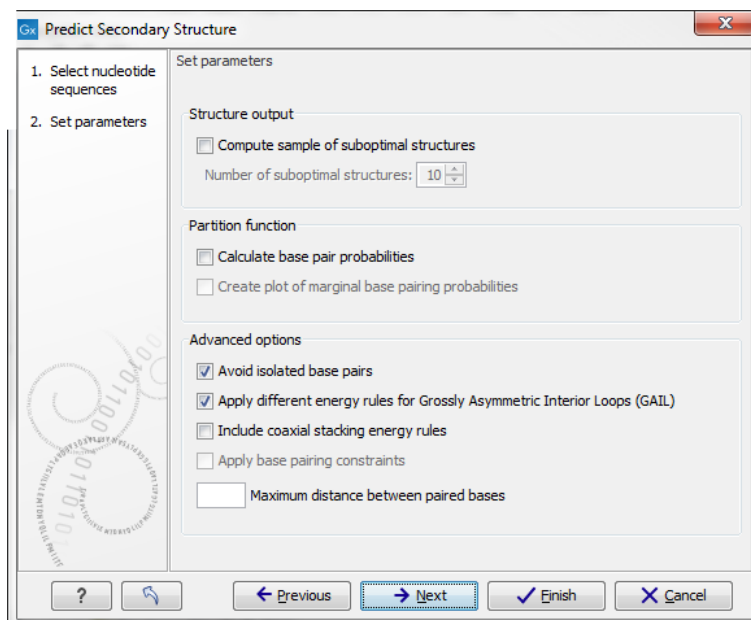
Figure 31.2: *Adjusting parameters for secondary structure prediction.*

31.1.2 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].

31.1.3 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 Main 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 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 31.3.
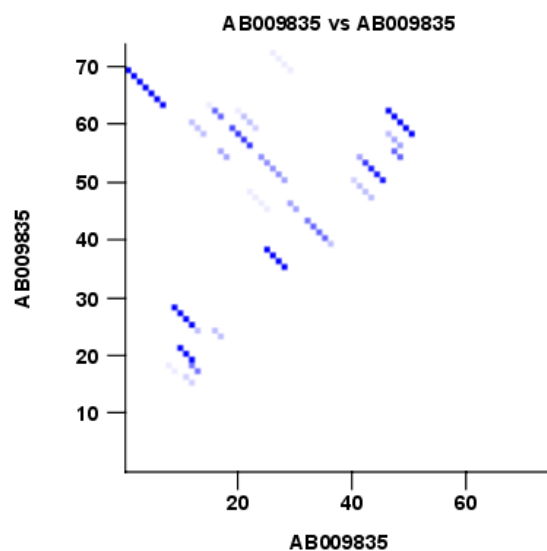
Figure 31.3: *The marginal base pair probability of all possible base pairs.*

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 31.2.3
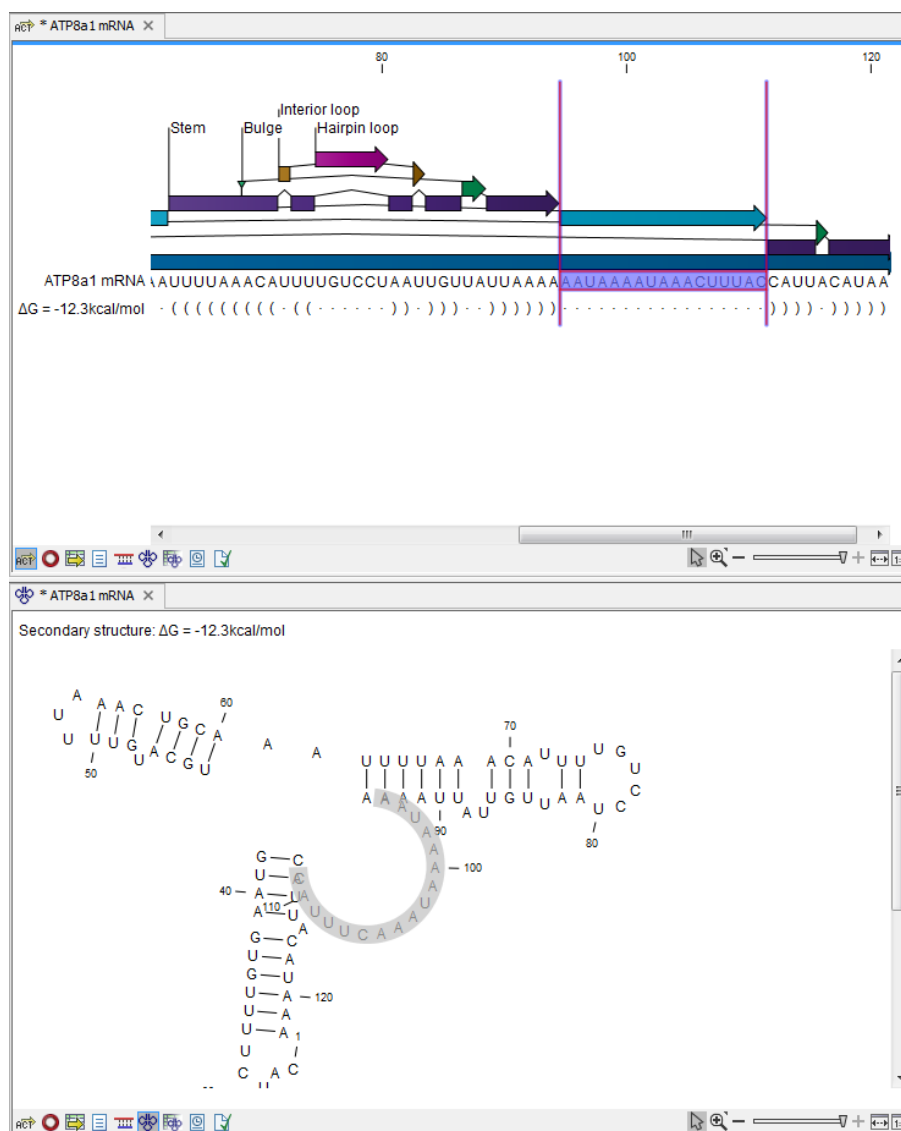
Figure 31.4: *Marginal probability of base pairs shown in linear view (top) and marginal probability of being unpaired shown in the secondary structure 2D view (bottom).*

and also in the secondary structure view. An example is shown in figure 31.4. Furthermore, the marginal probabilities are accessible from tooltips when hovering over the relevant parts of the structure.

31.1.4 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`).

- **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 Main 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 31.1.3).

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 31.5).
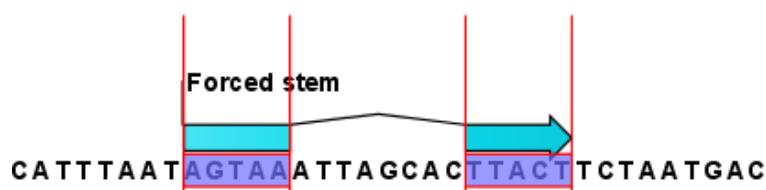
Figure 31.5: *Force a stem of the selected bases.*

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 31.6).
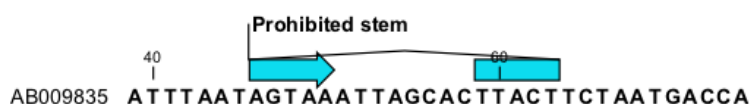
Figure 31.6: *Prohibit the selected bases from forming a stem.*

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 31.7.
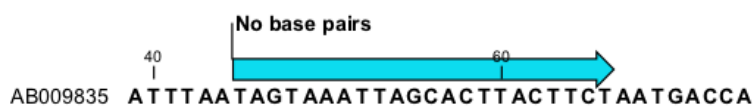
Figure 31.7: *Prohibiting any of the selected base from pairing with other bases.*

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.

31.1.5 Structure as annotation

You can choose to add the elements of the best structure as annotations (see figure 31.8).
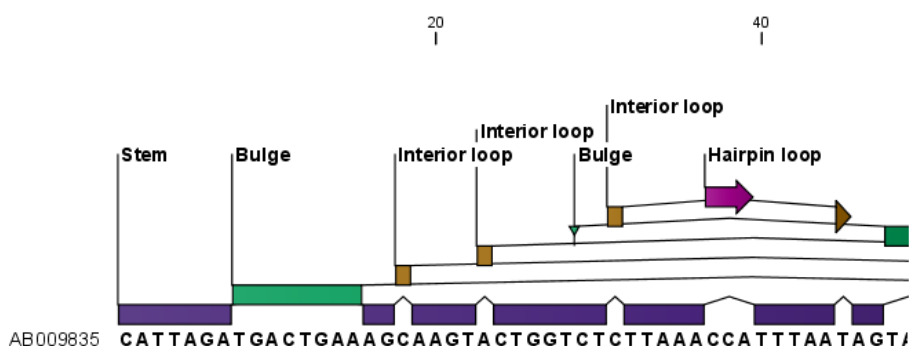
Figure 31.8: *Annotations added for each structure element.*

This makes it possible to use the structure information in other analysis in the *CLC Main 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 31.2.2.

31.2   View and edit secondary structures

When you predict RNA secondary structure (see section 31.1), the resulting predictions are attached to the sequence and can be shown as:

- Annotations in the ordinary sequence views (Linear sequence view (ᴀᴄᴛ→), Annotation table (🖼) etc. This is only possible if this has been chosen in the dialog in figure 31.2. See an example in figure 31.8.

- Symbolic representation below the sequence (see section 31.2.3).

- A graphical view of the secondary structure (see section 31.2.1).

- 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 31.2.2.

31.2.1   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 31.9.

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 31.9. 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 17.1.1. 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 31.2.2 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.
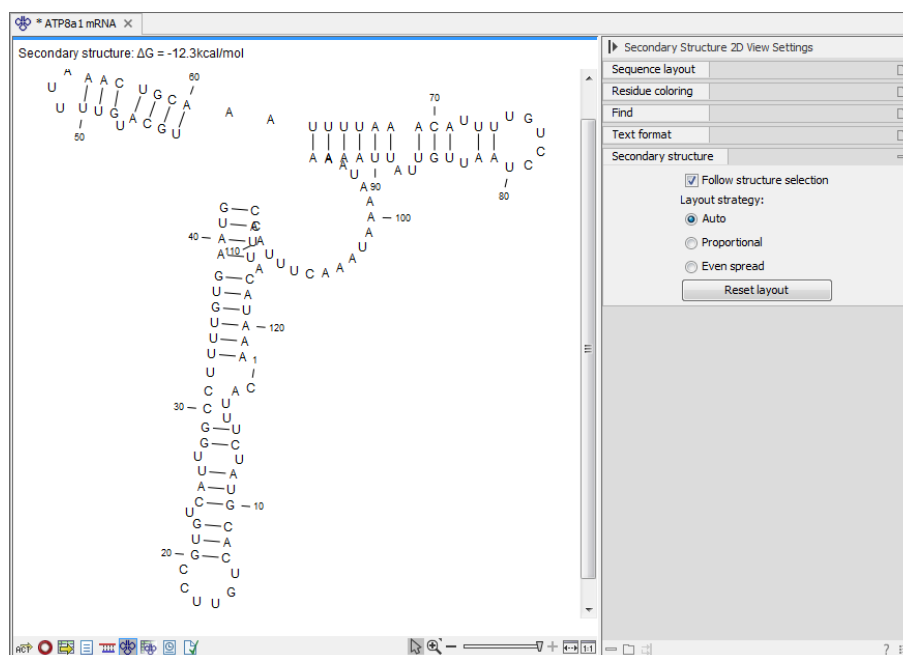
Figure 31.9: *The secondary structure view of an RNA sequence zoomed in.*

- **Auto.** The layout is adjusted to minimize overlapping structure elements [Han et al., 1999]. This is the default setting (see figure 31.10).

- **Proportional.** Arc lengths are proportional to the number of residues (see figure 31.11). Nothing is done to prevent overlap.

- **Even spread.** Stems are spread evenly around loops as shown in figure 31.12.

- **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.
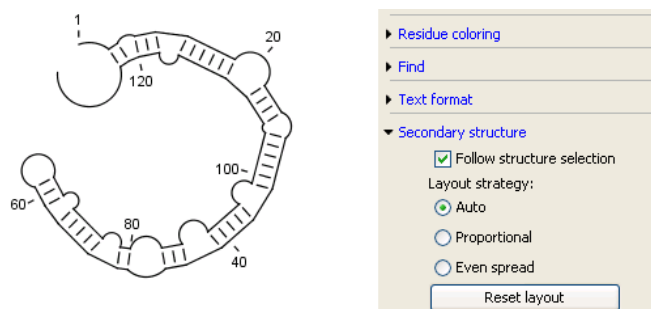
Figure 31.10: *Auto layout. Overlaps are minimized.*
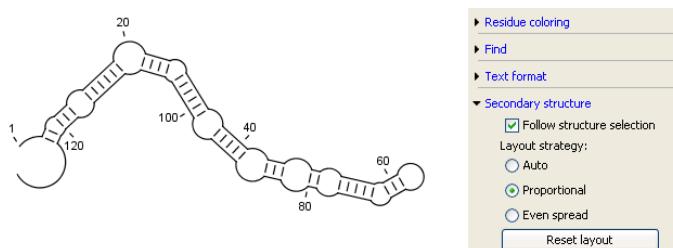
Figure 31.11: *Proportional layout. Length of the arc is proportional to the number of residues in the arc.*
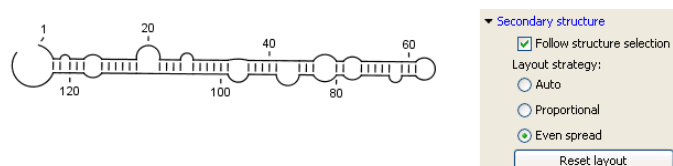
Figure 31.12: *Even spread. Stems are spread evenly around loops.*

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 31.13).
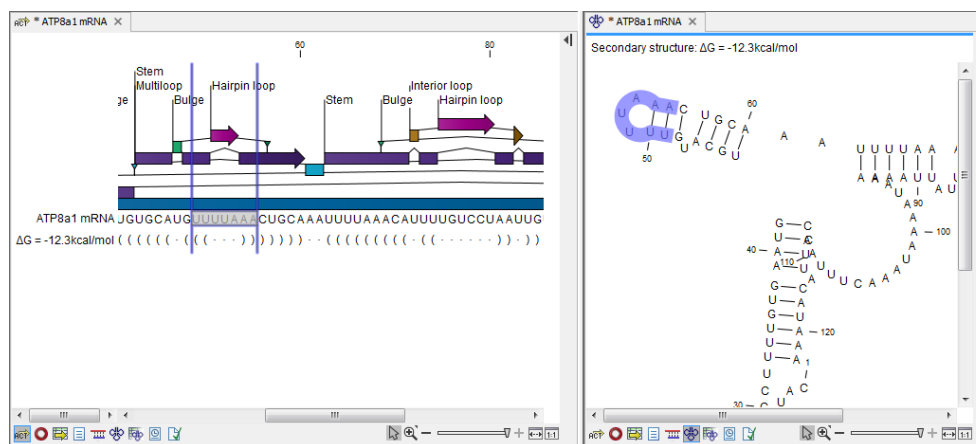
Figure 31.13: *A split view of the secondary structure view and a linear sequence view.*

If you make a selection in another sequence view, this will will also be reflected in the secondary structure view.

The *CLC Main 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 31.14).

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 31.15, the structure shown in figure 31.14 has been modified by dragging with the mouse.

Press **Reset layout** in the **Side Panel** to reset the layout to the way it looked when the structure
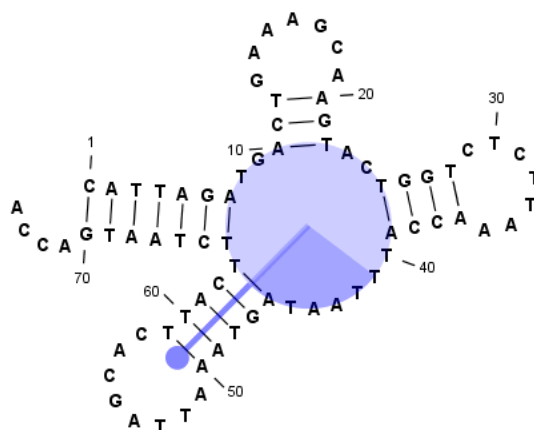
Figure 31.14: *The blue circle represents the anchor point for rotating the substructure.*
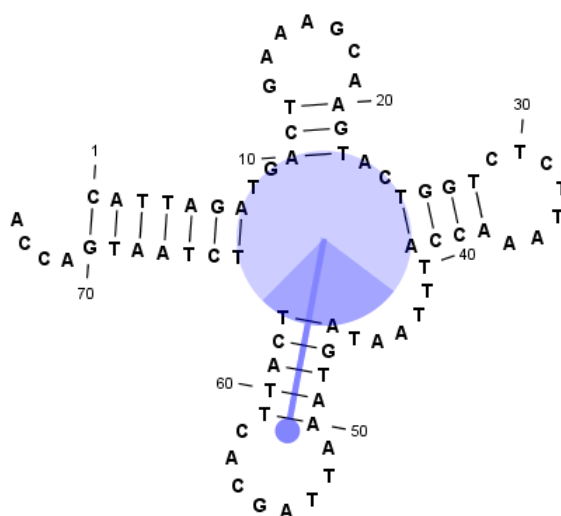
Figure 31.15: *The structure has now been rotated.*

was predicted.

31.2.2   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 31.1), 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 31.2.1).

- 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 31.16.

Figure 31.16: *The secondary structure table with the list of structures to the left, and to the right the substructures of the selected structure.*

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 31.5.2). Each substructure 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 31.17).

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
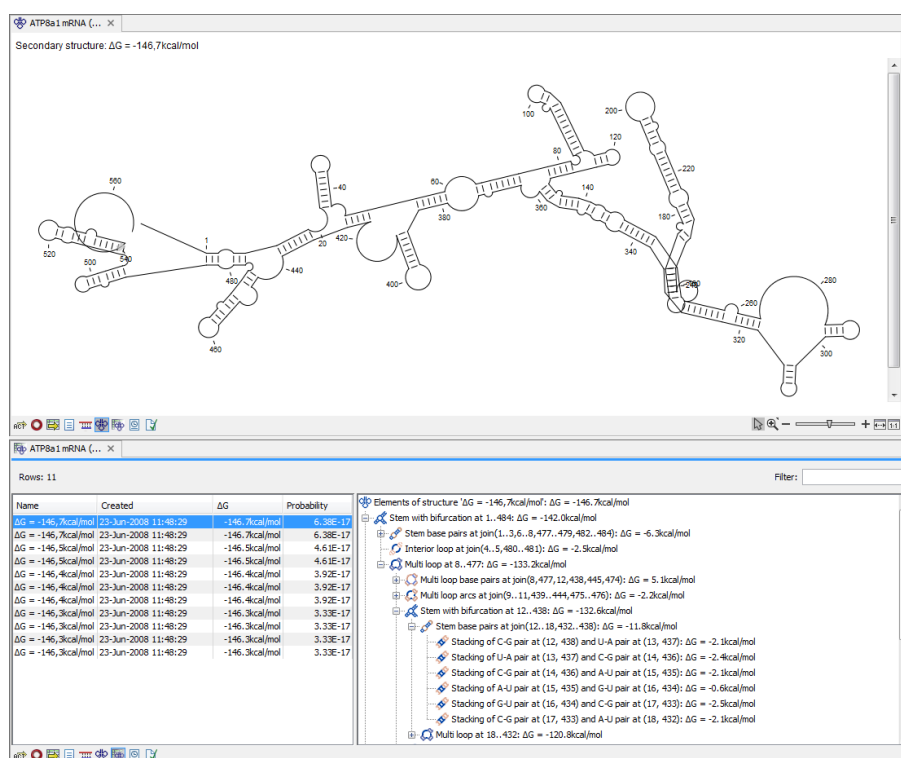
Figure 31.17: *A split view showing a structure table to the right and the secondary structure 2D view to the left.*

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 31.18).

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 31.18 the "Stem with bifurcation" is selected in the 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.
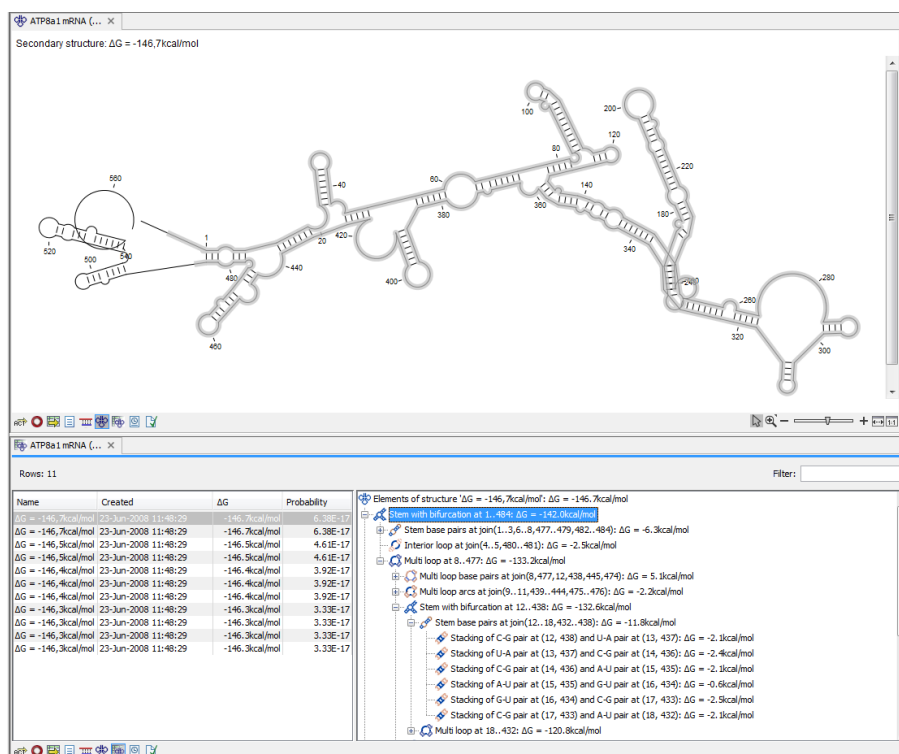
Figure 31.18: *Now the "Stem with bifurcation" node has been selected in the table and a corresponding selection has been made in the view of the secondary structure to the left.*

- **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.

31.2.3 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 31.19).

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.
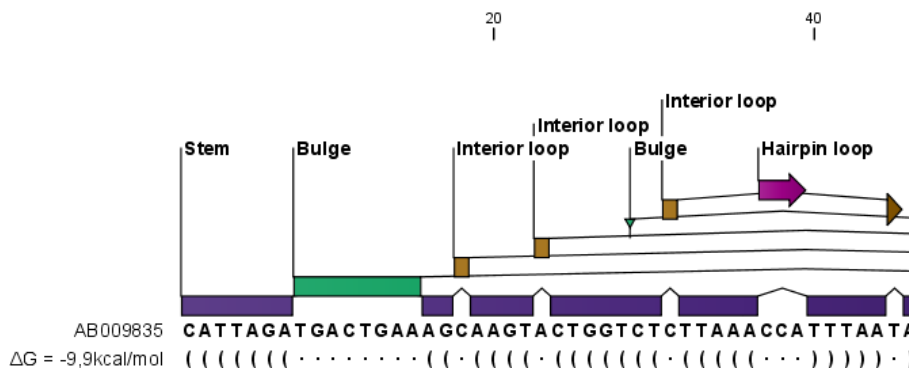
Figure 31.19: *The secondary structure visualized below the sequence and with annotations shown above.*

- 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.

- **Base pair symbol.** How a base pair should be represented (see figure 31.19).

- **Unpaired symbol.** How bases which are not part of a base pair should be represented (see figure 31.19).

- **Height.** When you zoom out, this option determines the height of the symbols as shown in figure 31.20 (when zoomed in, there is no need for specifying the height).

- **Base pair probability.** See section 31.2.4 below).

When you zoom in and out, the appearance of the symbols change. In figure 31.19, the view is zoomed in. In figure 31.20 you see the same sequence zoomed out to fit the width of the sequence.
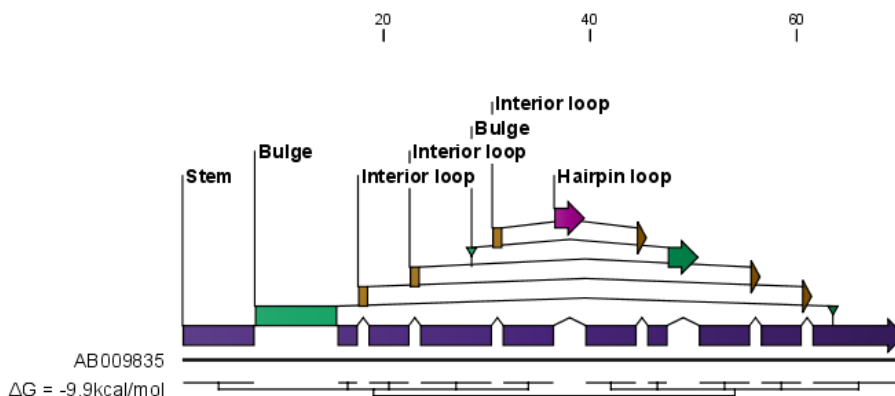
Figure 31.20: *The secondary structure visualized below the sequence and with annotations shown above. The view is zoomed out to fit the width of the sequence.*

31.2.4   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 31.4.

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.

31.3   Evaluate structure hypothesis

Hypotheses about an RNA structure can be tested using *CLC Main Workbench*. A structure hypothesis *H* is formulated using the structural constraint annotations described in section 31.1.4. By adding several annotations complex structural hypotheses can be formulated (see 31.21).

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:

$$P(H) = \frac{\displaystyle\sum_{s_H \in S_H} P(s_H)}{\displaystyle\sum_{s \in S} P(s)} = \frac{PF_H}{PF_{\text{full}}},$$

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. 31.21.
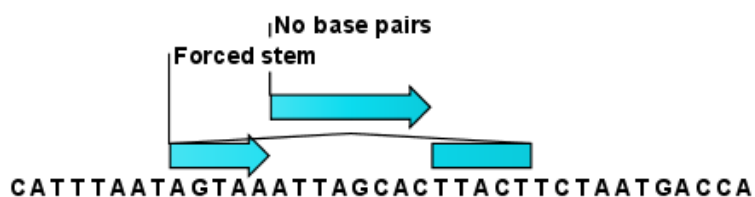
Figure 31.21: *Two constraints defining a structural hypothesis.*

31.3.1   Selecting sequences for evaluation

The evaluation is started from the **Toolbox**:

> **Toolbox | RNA Structure ( )| Evaluate Structure Hypothesis ( )**

This opens the dialog shown in figure 31.22.

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
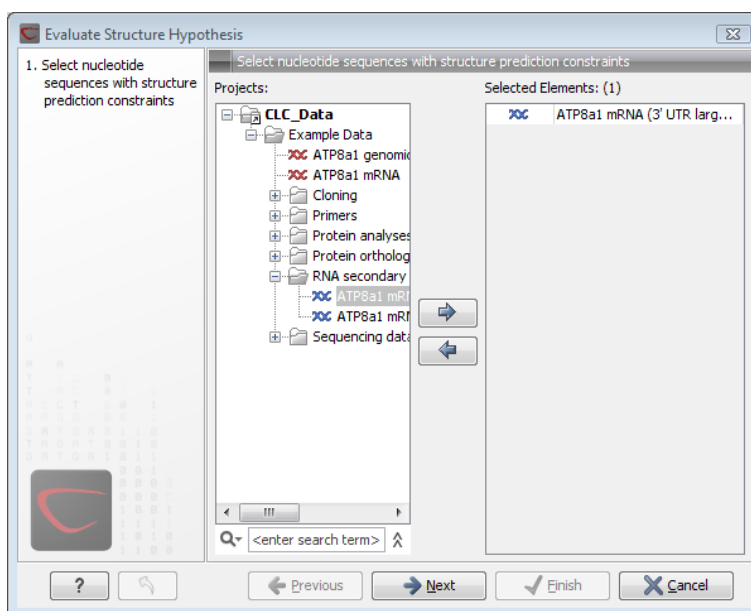
Figure 31.22: *Selecting RNA or DNA sequences for evaluating structure hypothesis.*

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 31.23).

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].

31.3.2   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 31.24).

31.4   Structure Scanning Plot

In *CLC Main 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
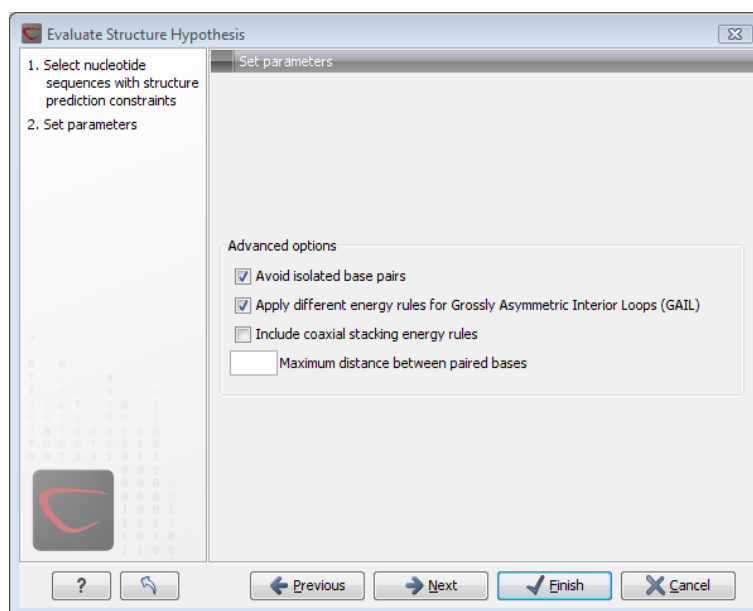
Figure 31.23: *Adjusting parameters for hypothesis evaluation.*
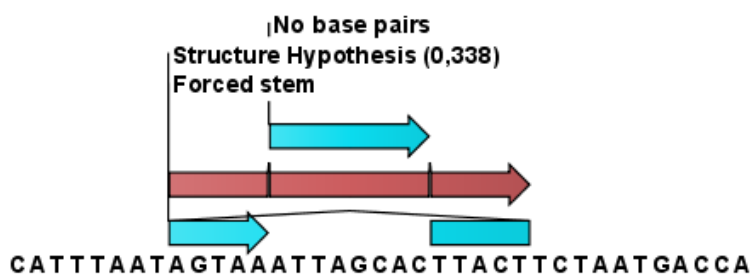
Figure 31.24: *This hypothesis has a probability of 0.338 as shown in the annotation.*

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].

31.4.1   Selecting sequences for scanning

The scanning is started from the **Toolbox**:

>   **Toolbox | RNA Structure ( )| Evaluate Structure Hypothesis ( )**

This opens the dialog shown in figure 31.25.

If you have selected sequences before choosing the Toolbox action, they are now listed in
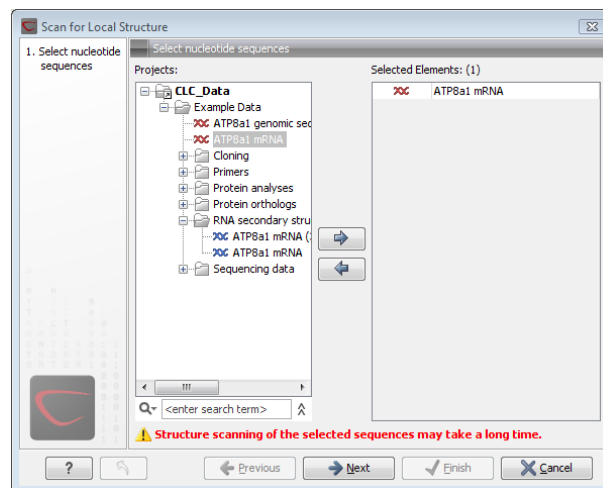
Figure 31.25: *Selecting RNA or DNA sequences for structure scanning.*

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 31.26).

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.
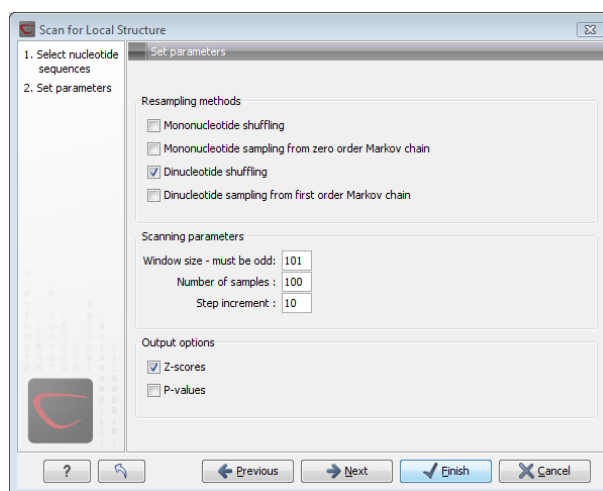
Figure 31.26: *Adjusting parameters for structure scanning.*

31.4.2   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 31.27).
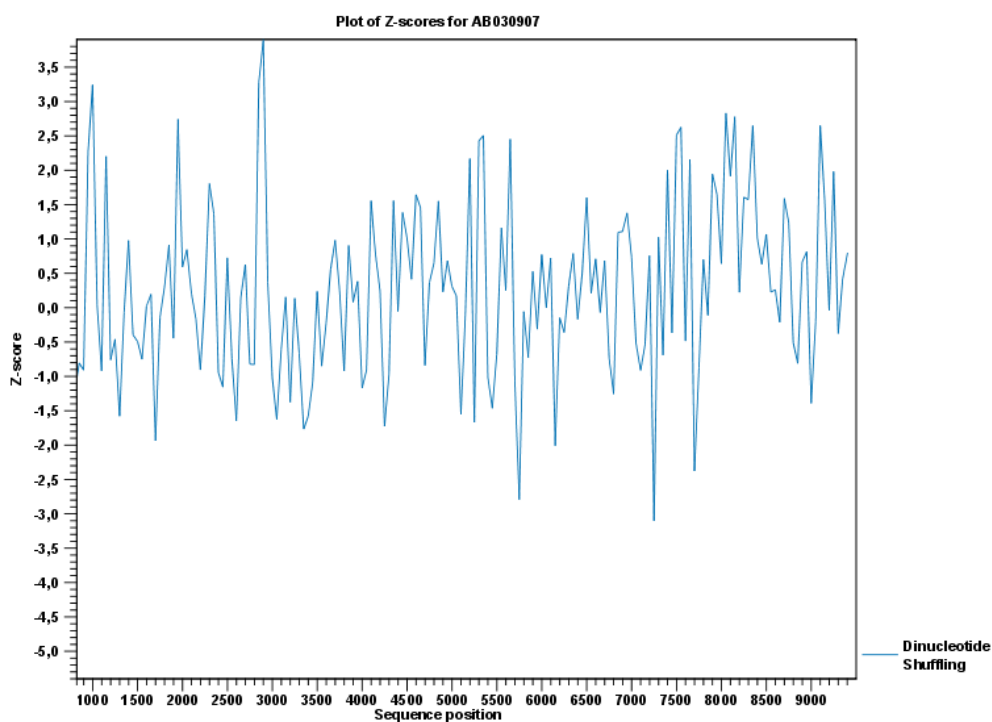
Figure 31.27: *A plot of the Z-scores produced by sliding a window along a sequence.*

31.5   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].

31.5.1   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 31.28.
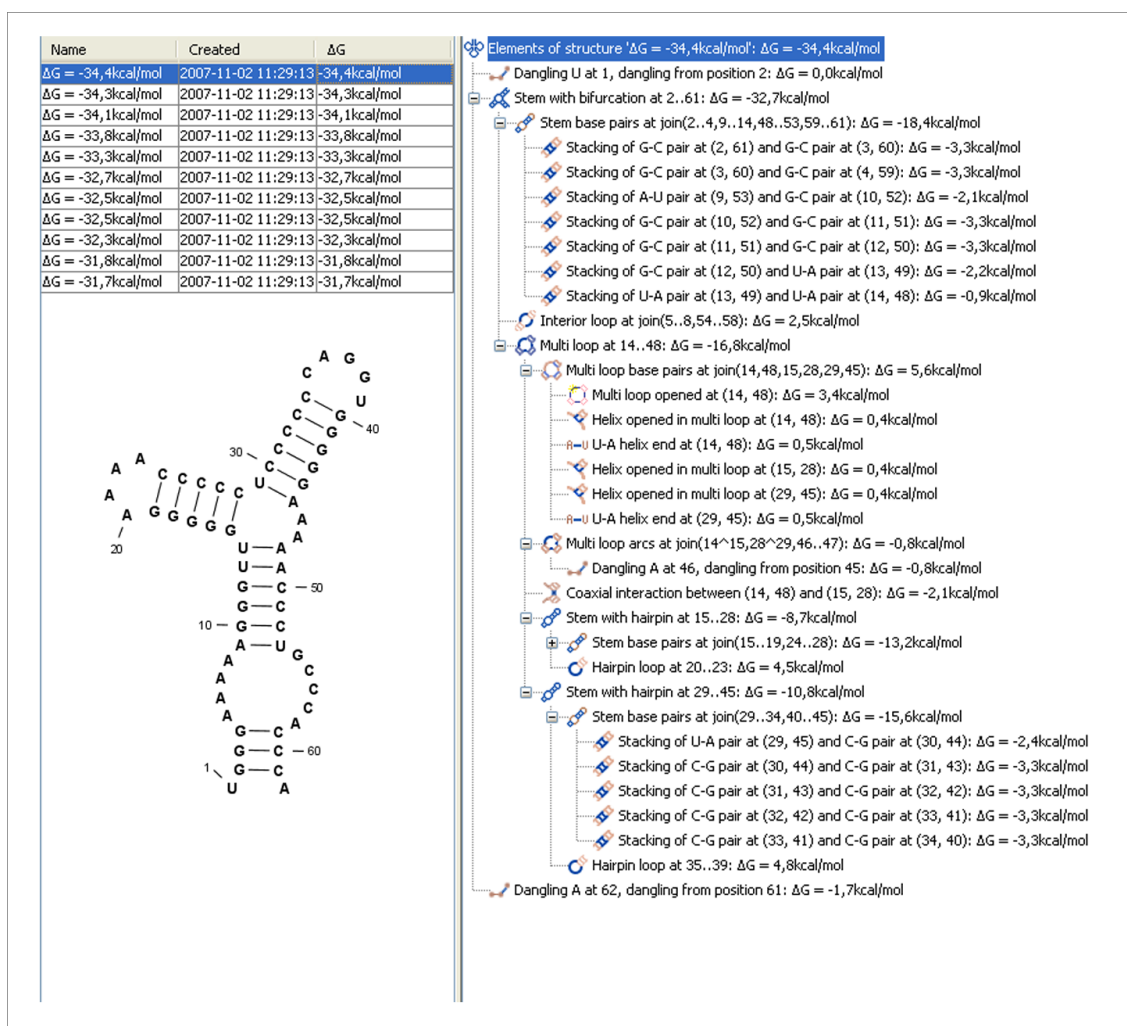
Figure 31.28: *A number of suboptimal structures have been predicted using* **CLC Main Workbench** *and are listed at the top left. At the right hand side, the structural components of the selected structure are listed in a hierarchical structure and on the left hand side the structure is displayed.*

31.5.2   Structure elements and their energy contribution

In this section, we classify the structure elements defining a secondary structure and describe their energy contribution.

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
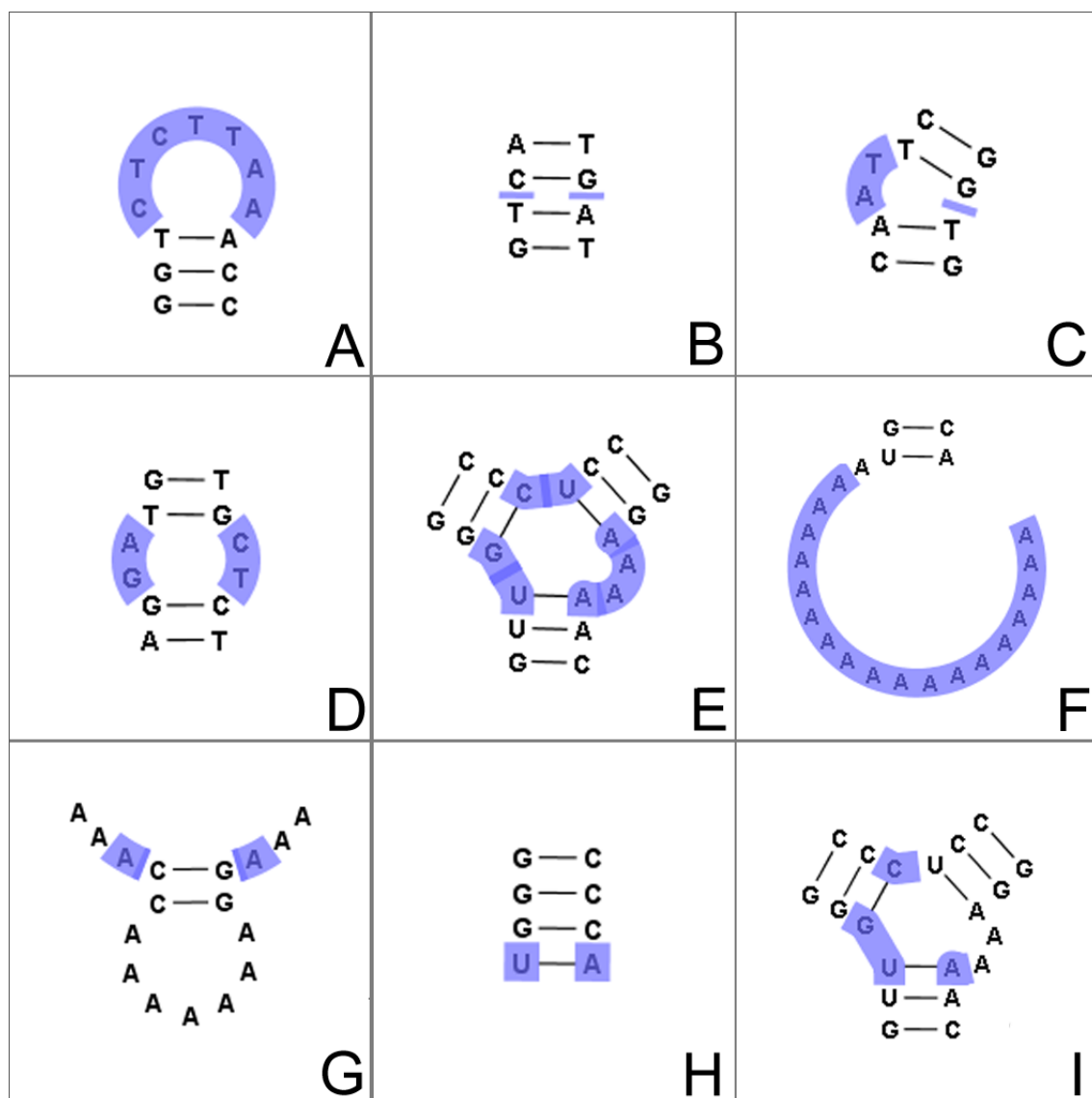
Figure 31.29: *The different structure elements of RNA secondary structures predicted with the free energy minimization algorithm in* **CLC Main Workbench**. *See text for a detailed description.*

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 31.29A).

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 31.29B). 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 31.29C). 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 31.29D). 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 31.29E). 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 31.29F). 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 31.29G). 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 31.29H).

- **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 31.29I). 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 31.30) and it has been shown that this can dramatically increase the fidelity of the secondary structure prediction [Mathews and Turner, 2006].

Creative Commons License

Figure 31.30: *Known structural features can be added as constraints to the secondary structure prediction algorithm in* **CLC Main Workbench**.

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 Main Workbench (■)

- CLC Genomics Workbench (■)

Data handling	Viewer	Main	Genomics
Add multiple locations to Navigation Area		■	■
Share data on network drive		■	■
Search all your data		■	■

Assembly of sequencing data	Viewer	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	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	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	Main	Genomics
Advanced molecular cloning		🟨	🟦
Graphical display of in silico cloning		🟨	🟦
Advanced sequence manipulation		🟨	🟦

Database searches	Viewer	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	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 report		■	■

Nucleotide analyses	Viewer	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 and alignments		■	■
G/C content analyses and graphs		■	■

Protein analyses	Viewer	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	Main	Genomics
Multiple sequence alignments (Two algorithms)	■	■	■
Advanced re-alignment and fix-point alignment 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	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	Main	Genomics
Dot plot based analyses		■	■

Phylogenetic trees	Viewer	Main	Genomics
Neighbor-joining and UPGMA phylogenies	■	■	■
Maximum likelihood phylogeny of nucleotides		■	■

Pattern discovery	Viewer	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	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 probe design		■	■
Match primer with sequence		■	■
Ordering of primers		■	■
Advanced analysis of primer properties		■	■

Molecular cloning	Viewer	Main	Genomics
Advanced molecular cloning		■	■
Graphical display of in silico cloning		■	■
Advanced sequence manipulation		■	■

Virtual gel view	Viewer	Main	Genomics
Fully integrated virtual 1D DNA gel simulator		■	■

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.

656

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 11.6).

Appendix C

BLAST databases

Several databases are available at NCBI, which can be selected to narrow down the possible BLAST hits.

C.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..

C.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.

C.3   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_BlastNucleotideDatabases.zip`

- Protein databases: `http://www.clcbio.com/wbsettings/NCBI_BlastProteinDatabases.zip`

Open the file you have downloaded into the `settings` folder, e.g. `NCBI_BlastProteinDatabases.properties` 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 D

Proteolytic cleavage enzymes

Most proteolytic enzymes cleave at distinct patterns. Below is a compiled list of proteolytic enzymes used in *CLC Main 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 E

Restriction enzymes database configuration

CLC Main 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: `http://www.clcbio.com/wbsettings/link_emboss_e_custom`. 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.

Please note that the CLC Workbenches only support the addition of 2-cutter enzymes. Further details about how to format your entries accordingly are given within the file mentioned above.

After adding the above file, or making changes to it, you must restart the Workbench for changes take effect.

Appendix F

Technical information about modifying Gateway cloning sites

The *CLC Main 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 F.1).
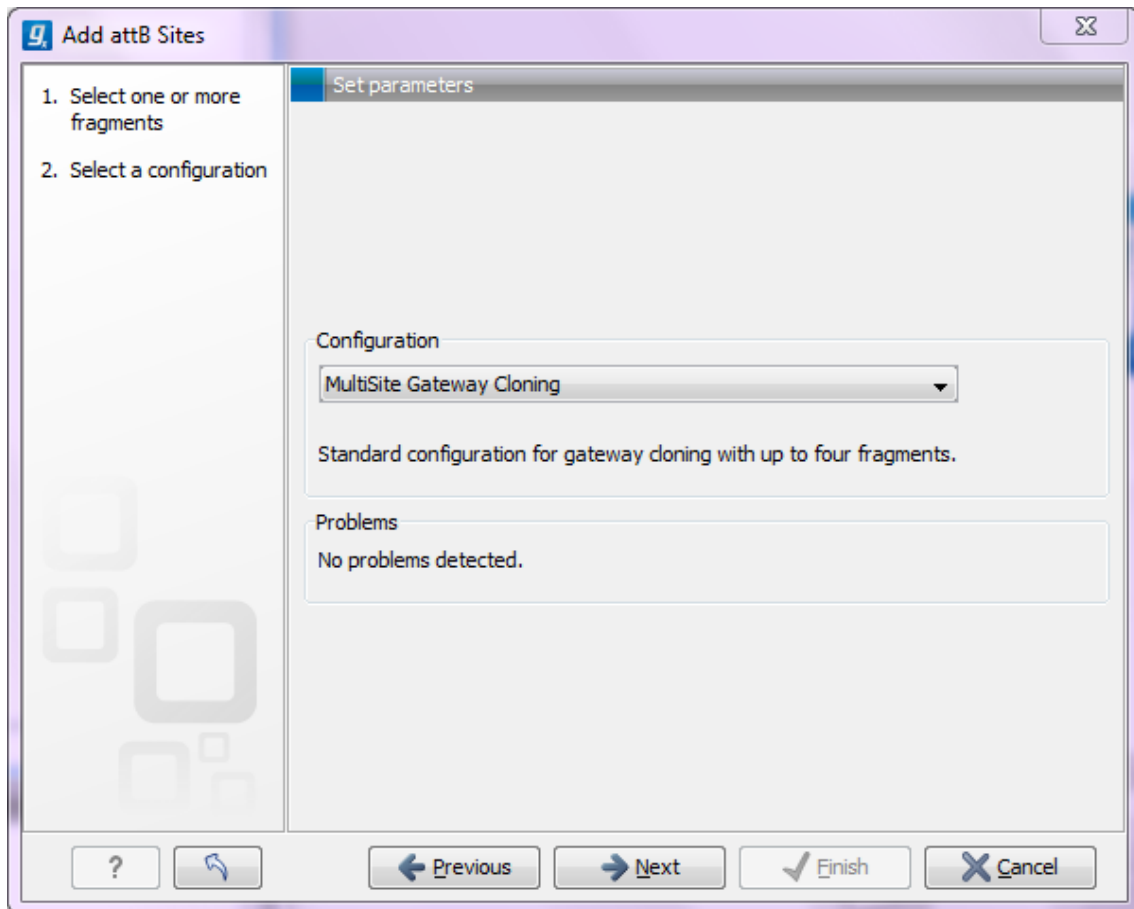
Figure F.1: *Selecting between different gateway cloning configurations.*

Appendix G

IUPAC codes for amino acids

(Single-letter codes based on International Union of Pure and Applied Chemistry)

The information is gathered from: http://www.insdc.org/documents/feature_table.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 H

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.insdc.org/documents/feature_table.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 I

Formats for import and export

I.1   List of bioinformatic data formats

Below is a list of bioinformatic data formats, i.e. formats for importing and exporting molecule structures, sequences, alignments and trees.

I.1.1   Sequence data formats

File type	Suffix	Import	Export	Description
AB1	.ab1	X		Including chromatograms
ABI	.abi	X		Including chromatograms
CLC	.clc	X	X	Rich format including all information
CSV export	.csv		X	Annotations in csv format
DNAstrider	.str/.strider	X	X	
DS Gene	.bsml	X		
EMBL	.embl	X	X	Rich information incl. annotations (nucs only)
FASTA	.fsa/.fasta	X	X	Simple format, name & description
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	X	Including chromatograms
Sequence Comma separated values	.csv	X	X	Simple format. One seq per line: name, description(optional), sequence
Staden	.sdn	X		
Swiss-Prot	.swp	X	X	Rich information incl. annotations (only peptides)
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

*Vector NTI import functionality comes as standard within the CLC Main Workbench and can be installed as a plugin via the Plugins Manager of the CLC Genomics Workbench (read more in section 1.7.1).

When exporting in fasta format, it is possible to remove sequence ends covered by annotations

of type "Trim" (read more in section 26.3).

I.1.2   Contig formats

File type	Suffix	Import	Export	Description
ACE	.ace	X	X	No chromatogram or quality score
CLC	.clc	X	X	Rich format including all information

I.1.3   Alignment formats

File type	Suffix	Import	Export	Description
Aligned fasta	.fa	X	X	Simple fasta-based format with – for gaps
CLC	.clc	X	X	Rich format including all information
ClustalW	.aln	X	X	
GCG Alignment	.msf	X	X	
Nexus	.nxs/.nexus	X	X	
Phylip Alignment	.phy	X	X	

I.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	

I.1.5   Expression data formats

Read about technical details of these data formats in section J.

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

I.1.6 Other formats

File type	Suffix	Import	Export	Description
CLC	.clc	X	X	Rich format including all information
mmCIF	.cif	X		3D structure
PDB	.pdb	X		3D structure
RNA structures	.ct, .col, .rnaml/.xml	x		Secondary structure for RNA

I.1.7 Table and text formats

File type	Suffix	Import	Export	Description
CSV	.csv		X	All tables
Excel	.xls/.xlsx		X	All tables and reports
Tab delimited	.txt		X	All tables
Text	.txt	X	X	All data in a textual format

Please see table I.1.5 **Expression data formats** for special cases of table imports.

I.1.8 File compression formats

File type	Suffix	Import	Export	Description
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gz/.tar	X		Contained files/folder structure (.tar and .zip not supported for NGS data)

Note! It is possible to import 'external' files into the Workbench and view these in the **Navigation Area**, but it is only the above mentioned formats whose *contents* can be shown in the Workbench.

I.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 13.3 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 J

Microarray data formats

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 section J.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 J.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 section J.5).

Below you find descriptions of the microarray data formats that are supported by *CLC Main Workbench*. Note that we for some platforms support both expression data and annotation data.

J.1 GEO (Gene Expression Omnibus)

The GEO (Gene Expression Omnibus) sample and series formats are supported. Figure J.1 shows how to download the data from GEO in the right format. GEO is located at `http://www.ncbi.nlm.nih.gov/geo/`.

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
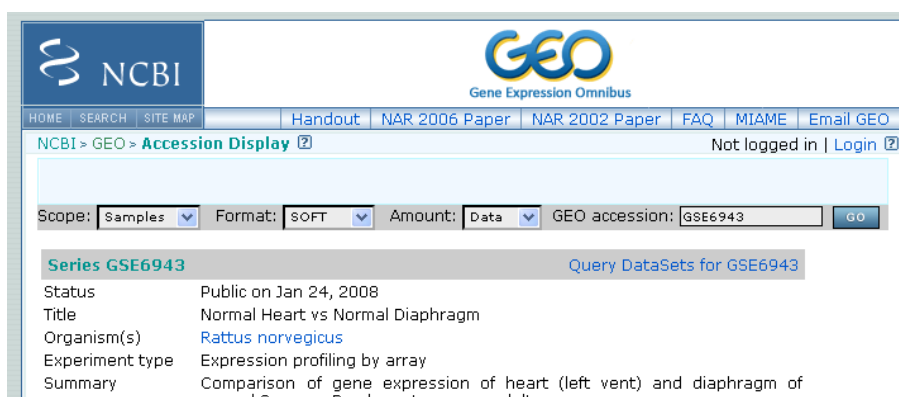
Figure J.1: *Selecting Samples, SOFT and Data before clicking go will give you the format supported by the* **CLC Main Workbench**.

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.

J.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

J.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

J.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

J.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

J.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

J.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 J.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.

J.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

J.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

J.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 27.1.3.

Download a small example annotation file here which includes header information:
http://www.clcbio.com/madata/AffymetrixNetAffxAnnotationFile.csv

J.3   Illumina BeadChip

Both BeadChip expression data files from Illumina's BeadStudio software and the corresponding BeadChip annotation files are supported by *CLC Main Workbench*. The formats of the BeadStudio

and annotation files have changed somewhat over time and various formats are supported.

J.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

J.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

J.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

J.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.

J.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 27.1.3.

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

J.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 Main Workbench*. This can be used to annotate experiments as shown in section 27.1.3. 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.

J.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 Main 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.

J.5.1   Generic expression data table format

The *CLC Main 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.

3. the file contains at least two samples.

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

J.5.2 Generic annotation file for expression data format

The *CLC Main 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.

When adding annotations to an experiment, you can specify the column in your annotation file containing the relevant identifiers. These identifiers are matched to the feature ids already present in your experiment. When a match is found, the annotation is added to that entry in the experiment. In other words, at least one column in your annotation file must contain identfiers matching the feature identifiers in the experiment, for those annotations to be applied.

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 component
Gene Ontology Molecular Function, Ontology_Function, GO_molecular_function	GO molecular function	Gene Ontology molecular function
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 K

Custom codon frequency tables

You can edit the list of codon frequency tables used by *CLC Main 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. In existing tables, the "_number" at the end of the ".cftbl" file name is the number of CDSs that were used for calculation, according to the `http://www.kazusa.or.jp/codon/` site.

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

[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.

[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.

[Bishop and Friday, 1985] Bishop, M. J. and Friday, A. E. (1985). Evolutionary trees from nucleic acid and protein sequences. *Proceeding of the Royal Society of London*, B 226:271–302.

[Blaisdell, 1989] Blaisdell, B. E. (1989). Average values of a dissimilarity measure not requiring sequence alignment are twice the averages of conventional mismatch counts requiring sequence alignment for a computer-generated model system. *J Mol Evol*, 29(6):538–47.

[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.

[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.

[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.

[Dayhoff et al., 1978] Dayhoff, M. O., Schwartz, R. M., and Orcutt, B. C. (1978). A model of evolutionary change in protein. *Atlas of Protein Sequence and Structure*, 5(3):345–352.

[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.

[Edgar, 2004] Edgar, R. C. (2004). Muscle: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113.

[Efron, 1982] Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans*, volume 38. SIAM.

[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.

[Felsenstein, 1985] Felsenstein, J. (1985). Confidence limits on phylogenies: An approach using the bootstrap. *Journal of Molecular Evolution*, 39:783–791.

[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.

[Gentleman and Mullin, 1989] Gentleman, J. F. and Mullin, R. (1989). The distribution of the frequency of occurrence of nucleotide subsequences, based on their overlap capability. *Biometrics*, 45(1):35–52.

[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.

[Höhl et al., 2007] Höhl, M., Rigoutsos, I., and Ragan, M. A. (2007). Pattern-based phylogenetic distance estimation and tree reconstruction. *Evolutionary Bioinformatics*, 2:0–0.

[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.

[Jones et al., 1992] Jones, D., Taylor, W., and Thornton, J. (1992). The rapid generation of mutation data matrices from protein sequences. *Computer Applications in the Biosciences (CABIOS)*, 8:275–282.

[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.

[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.

[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.

[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.

[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.

[Robinson et al., 2010] Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140.

[Robinson and Smyth, 2007] Robinson, M. D. and Smyth, G. K. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887.

[Robinson and Smyth, 2008] Robinson, M. D. and Smyth, G. K. (2008). Small-sample estimation of negative binomial dispersion, with applications to sage data. *Biostatistics*, 9(2):321–332.

[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.

[Sturges, 1926] Sturges, H. A. (1926). The choice of a class interval. *Journal of the American Statistical Association*, 21:65–66.

[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.

[Whelan and Goldman, 2001] Whelan, S. and Goldman, N. (2001). A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. *Molecular Biology and Evolution*, 18:691–699.

[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.

[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.

[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