



CLC **Main** Workbench

USER MANUAL

Manual for
CLC Main Workbench 8.1
Windows, macOS and Linux

December 7, 2018

This software is for research purposes only.

QIAGEN Aarhus
Silkeborgvej 2
Prismet
DK-8000 Aarhus C
Denmark



Contents

I	Introduction	10
1	Introduction to CLC Main Workbench	11
1.1	Contact information	12
1.2	Download and installation	12
1.3	System requirements	15
1.4	Workbench Licenses	16
1.5	Plugins	32
1.6	Network configuration	35
1.7	CLC Server connection	36
1.8	Getting started and latest improvements	37
II	Core Functionalities	39
2	User interface	40
2.1	View Area	41
2.2	Zoom and selection in View Area	49
2.3	Toolbox and Status Bar	51
2.4	Workspace	55
2.5	List of shortcuts	56
3	Data management and search	59
3.1	Navigation Area	60
3.2	Metadata	69
3.3	Working with tables	84
3.4	Customized attributes on data locations	87

3.5	Local search	92
4	User preferences and settings	98
4.1	General preferences	98
4.2	View preferences	100
4.3	Data preferences	103
4.4	Advanced preferences	103
4.5	Export/import of preferences	104
4.6	View settings for the Side Panel	105
5	Printing	107
5.1	Selecting which part of the view to print	108
5.2	Page setup	109
5.3	Print preview	111
6	Import/export of data and graphics	112
6.1	Standard import	112
6.2	Data export	114
6.3	Export graphics to files	124
6.4	Export graph data points to a file	129
6.5	CLC Server data import and export	130
6.6	Copy/paste view output	130
7	Data download	132
7.1	Search for Sequences at NCBI	132
7.2	Search for PDB Structures at NCBI	134
7.3	Search for Sequences in UniProt (Swiss-Prot/TrEMBL)	138
7.4	Sequence web info	141
8	Running tools, handling results and batching	143
8.1	Running tools	143
8.2	Handling results	145
8.3	Batch processing	147
9	Workflows	151

9.1	Creating a workflow	152
9.2	Distributing and installing workflows	173
9.3	Executing a workflow	182
9.4	Open copy of installed workflow	183
9.5	Batch launching workflows with multiple inputs	183
10	Other data types	189
10.1	Tracks	189
III	Bioinformatics	190
11	Viewing and editing sequences	191
11.1	View sequence	191
11.2	Circular DNA	201
11.3	Working with annotations	203
11.4	Element information	212
11.5	View as text	213
11.6	Sequence Lists	214
12	3D Molecule Viewer	217
12.1	Importing molecule structure files	219
12.2	Viewing molecular structures in 3D	222
12.3	Customizing the visualization	224
12.4	Tools for linking sequence and structure	233
12.5	Protein structure alignment	236
13	Sequence alignment	241
13.1	Create an alignment	241
13.2	View alignments	246
13.3	Edit alignments	250
13.4	Join alignments	254
13.5	Pairwise comparison	255
14	Phylogenetic trees	261

14.1	K-mer Based Tree Construction	263
14.2	Create tree	264
14.3	Model Testing	265
14.4	Maximum Likelihood Phylogeny	267
14.5	Tree Settings	276
14.6	Metadata and phylogenetic trees	285
15	General sequence analyses	290
15.1	Extract Annotations	290
15.2	Extract sequences	292
15.3	Shuffle sequence	294
15.4	Dot plots	295
15.5	Local complexity plot	304
15.6	Sequence statistics	304
15.7	Join sequences	309
15.8	Pattern discovery	310
15.9	Motif Search	312
15.10	Create motif list	317
16	Nucleotide analyses	319
16.1	Convert DNA to RNA	319
16.2	Convert RNA to DNA	320
16.3	Reverse complements of sequences	320
16.4	Reverse sequence	321
16.5	Translation of DNA or RNA to protein	322
16.6	Find open reading frames	323
17	Protein analyses	327
17.1	Protein charge	327
17.2	Antigenicity	329
17.3	Hydrophobicity	330
17.4	Pfam domain search	334
17.5	Secondary structure prediction	337

17.6	Protein report	338
17.7	Reverse translation from protein into DNA	340
17.8	Proteolytic cleavage detection	343
18	Sequencing data analyses and Assembly	348
18.1	Importing and viewing trace data	349
18.2	Trim sequences	350
18.3	Assemble sequences	353
18.4	Assemble sequences to reference	355
18.5	Sort sequences by name	357
18.6	Add sequences to an existing contig	361
18.7	View and edit contigs and read mappings	362
18.8	Reassemble contig	371
18.9	Secondary peak calling	371
19	Primers and probes	373
19.1	Primer design - an introduction	374
19.2	Setting parameters for primers and probes	376
19.3	Graphical display of primer information	378
19.4	Output from primer design	380
19.5	Standard PCR	382
19.6	Nested PCR	385
19.7	TaqMan	387
19.8	Sequencing primers	388
19.9	Alignment-based primer and probe design	389
19.10	Analyze primer properties	394
19.11	Find binding sites and create fragments	395
19.12	Order primers	399
20	Cloning and restriction sites	401
20.1	Restriction site analyses	401
20.2	Restriction enzyme lists	410
20.3	Molecular cloning	411

20.4	Gateway cloning	420
20.5	Gel electrophoresis	426
21	RNA structure	430
21.1	RNA secondary structure prediction	431
21.2	View and edit secondary structures	437
21.3	Evaluate structure hypothesis	445
21.4	Structure scanning plot	446
21.5	Bioinformatics explained: RNA structure prediction by minimum free energy minimization	448
22	Expression analysis	455
22.1	Experimental design	456
22.2	Transformation and normalization	469
22.3	Quality control	473
22.4	Statistical analysis - identifying differential expression	486
22.5	Feature clustering	497
22.6	Annotation tests	505
22.7	General plots	511
23	BLAST search	518
23.1	Running BLAST searches	519
23.2	Output from BLAST searches	526
23.3	Extract consensus sequence	532
23.4	Local BLAST databases	534
23.5	Manage BLAST databases	537
23.6	Bioinformatics explained: BLAST	538
24	Utility Tools	547
24.1	Batch Rename	547
24.2	Extract Annotations	552

IV Appendix	554
A Graph preferences	555
B BLAST databases	557
B.1 Peptide sequence databases	557
B.2 Nucleotide sequence databases	557
B.3 Adding more databases	558
C Proteolytic cleavage enzymes	560
D Restriction enzymes database configuration	562
E Technical information about modifying Gateway cloning sites	563
F IUPAC codes for amino acids	564
G IUPAC codes for nucleotides	565
H Formats for import and export	566
H.1 List of bioinformatic data formats	566
H.2 List of graphics data formats	569
I Gene expression annotation files and microarray data formats	570
I.1 GEO (Gene Expression Omnibus)	570
I.2 Affymetrix GeneChip	573
I.3 Illumina BeadChip	575
I.4 Gene ontology annotation files	576
I.5 Generic expression and annotation data file formats	576
J Custom codon frequency tables	580
Bibliography	581

Part I

Introduction

Chapter 1

Introduction to *CLC Main Workbench*

Contents

1.1	Contact information	12
1.2	Download and installation	12
1.2.1	Program download	13
1.2.2	Installation on Microsoft Windows	13
1.2.3	Installation on macOS	13
1.2.4	Installation on Linux with an installer	14
1.3	System requirements	15
1.4	Workbench Licenses	16
1.4.1	Request an evaluation license	17
1.4.2	Download a license using a license order ID	19
1.4.3	Import a license from a file	21
1.4.4	Upgrade license	22
1.4.5	Configure license server connection	25
1.4.6	Download a static license on a non-networked machine	29
1.4.7	Viewing mode	31
1.4.8	Start in safe mode	32
1.5	Plugins	32
1.5.1	Install	33
1.5.2	Uninstall	34
1.5.3	Updating plugins	34
1.6	Network configuration	35
1.7	CLC Server connection	36
1.7.1	CLC Server Connection	36
1.8	Getting started and latest improvements	37

Welcome to *CLC Main Workbench 8.1* – 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:

QIAGEN Aarhus
Silkeborgvej 2
Prismet
8000 Aarhus C
Denmark

<http://www.qiagenbioinformatics.com>

Email: ts-bioinformatics@qiagen.com

The QIAGEN Aarhus 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. **Getting help via the workbench**

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, Upgrades and Support (MUS) program (<http://www.qiagenbioinformatics.com/maintenance-and-support/>), you can contact our customer support via the workbench by going to the menu option:

Help | Contact Support

This will open a dialog where you can enter your contact information, and a text field for writing the question or problem you have. On a second dialog you will be given the chance to attach screenshots or even small datasets that can help explain or troubleshoot the problem. When you send a support request this way, it will automatically include helpful technical information about your installation and your license information so that you do not have to look this up yourself. Our support staff will reply to you by email.

Other ways to contact the support team

You can also contact the support team by email: ts-bioinformatics@qiagen.com

Please provide your contact information, your license information, some technical information about your installation, and describe the question or problem you have. You can also attach screenshots or even small data sets that can help explain or troubleshoot the problem.

Information about how to find your license information is included in the licenses section of our Frequently Asked Questions (FAQ) area: <https://secure.clcbio.com/helpspot/index.php?pg=kb>. Information about MUS cover on particular licenses can be found by <https://secure.clcbio.com/myclc/login>.

1.2 Download and installation

The *CLC Main Workbench* is developed for Windows, macOS and Linux. The software for either platform can be downloaded from <http://www.qiagenbioinformatics.com/product-downloads/>. To check for available updates of the workbench and plugins, click on **Help | Check for Updates...** (🔧).

1.2.1 Program 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

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.

1.2.2 Installation on Microsoft Windows

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 macOS

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**.
- 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_8_0_64.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:

```
# clcmainwb8
```

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

```
# ./clcmainwb8
```

1.3 System requirements

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

- Windows 7, Windows 8, Windows 10, Windows Server 2012, and Windows Server 2016
- OS X 10.10, 10.11 and macOS 10.12, 10.13, 10.14
- Linux: RHEL 6.7 and later, SUSE Linux Enterprise Server 11 and later. The software is expected to run without problem on other recent Linux systems, but we do not guarantee this.
- 64 bit operating system
- 1 GB RAM required
- 2 GB RAM recommended
- 1024 x 768 display required
- 1600 x 1200 display recommended

Special system requirements for the 3D Molecule Viewer

- **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 .
- **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.

Indirect rendering (such as x11 forwarding through ssh), remote desktop connection/VNC, and running in virtual machines is not supported.

1.4 Workbench Licenses

When you start up the *CLC Main Workbench* for the first time on your system, or after installing a new major release, the **License Assistant**, shown in figure 1.1, will be presented to you. The **License Assistant** can be also be launched during an active Workbench session by clicking on the "Upgrade Workbench License" button at the bottom of the **License Manager**. The **License Manager** can be started up using the Workbench menu item:

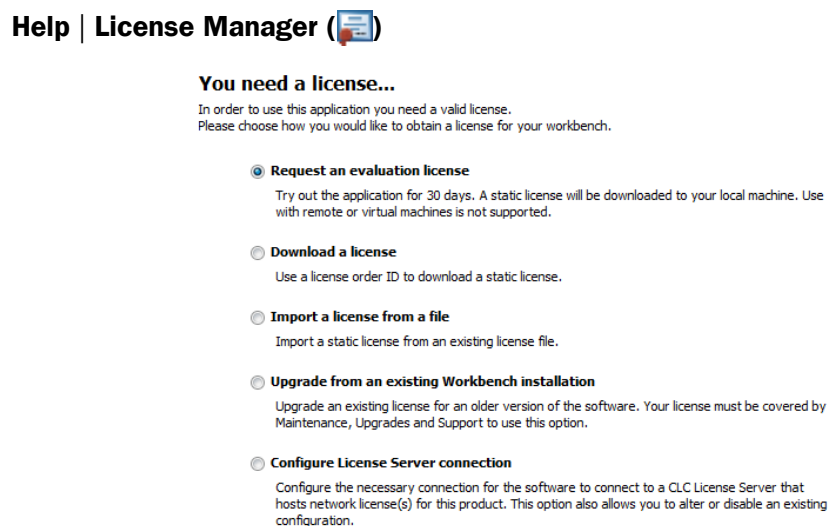


Figure 1.1: *The License Assistant provides access to licensing options.*

The options available in the **License Assistant** window are described in brief below, and then in detail in the sections that follow.

- **Request an evaluation license** Request a fully functional, time-limited license.
- **Download a license** Use the license order ID provided when you purchase the software to download and install a license file.
- **Import a license from a file** Import an existing license file, for example a file downloaded from the license download webpage.
- **Upgrade from an existing Workbench installation** If you have used a previous version of the *CLC Main Workbench*, and you are entitled to upgrade to a new major version, select this option to upgrade your license file.
- **Configure License Server connection** If your organization has a CLC License Server, select this option to configure the connection to it.

Select the appropriate option and then click on the **Next** button.

To use the **Request an evaluation license**, **Download a license** or the **Upgrade from an existing Workbench installation** options, your machine must be able to access the external network. If this is not the case, please see section 1.4.6.

When using a *CLC Main Workbench* installed in a central location on your system, you must be running the program in administrative mode to license the software. On Linux and Mac, this

means you must be logged in as an administrator. On Windows, you can right-click the program shortcut and choose "Run as Administrator".

If you do not have a license order ID or access to a license, you can still use the Workbench in **Viewing Mode**. See section 1.4.7) for further information about this.

1.4.1 Request an evaluation license

We offer a fully functional version of the *CLC Main Workbench* for evaluation purposes, free of charge. Each person is entitled to a 14-day trial of *CLC Main Workbench*. If you are unable to complete your assessment in the available time, please send an email to bioinformaticssales@qiagen.com to request an additional evaluation period.

When you choose the option **Request an evaluation license**, you will see the dialog shown in figure 1.2.

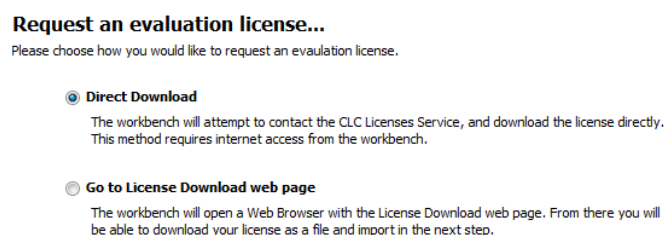


Figure 1.2: Choose between downloading a license directly, or opening the license download form in a web browser.

In this dialog, there are two options:

- **Direct Download.** Download the license directly. This method requires that the Workbench has access to the external network.
- **Go to CLC License Download web page.** The online license download form will be opened in a web browser. This option is suitable for when downloading a license for use on another machine that does not have access to the external network, and thus cannot access the QIAGEN Aarhus servers.

After selecting your method of choice, click on the button labeled **Next**.

Direct download

After choosing the **Direct Download** option and clicking on the button labeled **Next**, a dialog similar to that shown in figure 1.3 will appear if the license is successfully downloaded and installed.

When the license has been downloaded and installed, the **Next** button will be enabled.

If there is a problem, a dialog will appear indicating this.

Requesting a license...

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



An Evaluation License was successfully downloaded
The License is valid until: 2015-04-09

Figure 1.3: A license has been successfully downloaded and installed for use.

Go to license download web page

After choosing the **Go to CLC License Download web page** option and clicking on the button labeled **Next**, the license download form will be opened in a web browser, as shown in figure 1.4.

Download a license

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

You have requested the following license:

License Order-ID:	CLC-LICENSE-SRENMINSTED-0D43CA9EDF-40000D844A4C0C48000X		
Product:	1		
Product Version:	1		
Host-ID(s):	09EDE85CEFD	A81AEFF919F	2A378AC18863
Host name:	laptop-32		

To download your license, please click the button below.

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

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

Figure 1.4: The license download form opened in a web browser.

Click on the **Download License** button and then save the license file.

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

Import a license from a file...

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



No file selected...

Figure 1.5: Importing the license file downloaded from the web page.

Click on the **Choose License File** button, find the saved license file and select it. Then click on the **Next** button.

Accepting the license agreement

The end user license agreement (EULA) must be read and accepted as part of the installation process. figure 1.28.

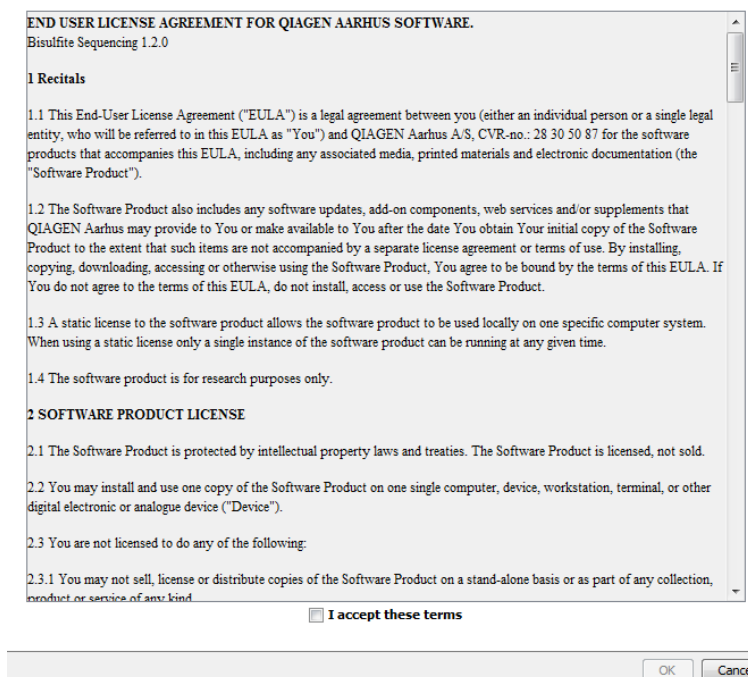


Figure 1.6: The End User License Agreement is presented during the installation process.

Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

1.4.2 Download a license using a license order ID

Using a license order ID, you can download a license file via the Workbench or using an online form. When you have chosen this option and clicked on the **Next** button, you will see the dialog shown in 1.7. Enter your license order ID into the License Order ID text field. (The ID can be pasted into the box after copying it and then right clicking in the text field and choosing Paste from the context menu, or using a key combination like Ctrl+V, or on a Mac, ⌘ + V).

Download a license...

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

License Order-ID:

CLC-LICENSE-XXXX-XXX-4B-G00-XXXXX-XXXXX

Direct Download

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

Go to License Download web page

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

Figure 1.7: Enter a license order ID into the text field and then click on the Next button.

In this dialog, there are two options:

- **Direct Download.** Download the license directly. This method requires that the Workbench has access to the external network.

- **Go to CLC License Download web page.** The online license download form will be opened in a web browser. This option is suitable for when downloading a license for use on another machine that does not have access to the external network, and thus cannot access the QIAGEN Aarhus servers.

After selecting your method of choice, click on the button labeled **Next**.

Direct download

After choosing the **Direct Download** option and clicking on the button labeled **Next**, a dialog similar to that shown in figure 1.8 will appear if the license is successfully downloaded and installed.

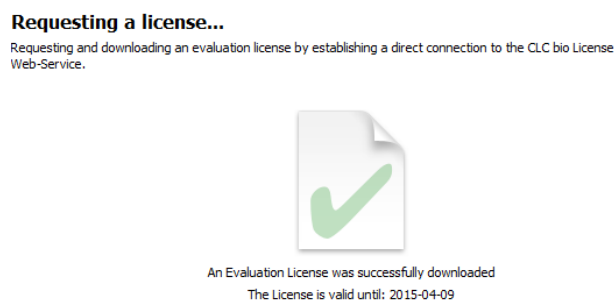


Figure 1.8: A license has been successfully downloaded and installed for use.

When the license has been downloaded and installed, the **Next** button will be enabled.

If there is a problem, a dialog will appear indicating this.

Go to license download web page

After choosing the **Go to CLC License Download web page** option and clicking on the button labeled **Next**, the license download form will be opened in a web browser, as shown in figure 1.9.

Download a license

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

You have requested the following license:

License Order-ID:	CLC-LICENSE-SRENMINSTED-0D43CA9EDF-60000D844A4C0C48XXXX
Product:	
Product Version:	1
Host-ID(s):	0SEDE85CEFD , A81AEFF919F , 2A378AC18863
Host name:	laptop-32

To download your license, please click the button below.

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

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

Figure 1.9: The license download form opened in a web browser.

Click on the **Download License** button and then save the license file.

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

Click on the **Choose License File** button, find the saved license file and select it. Then click on the **Next** button.

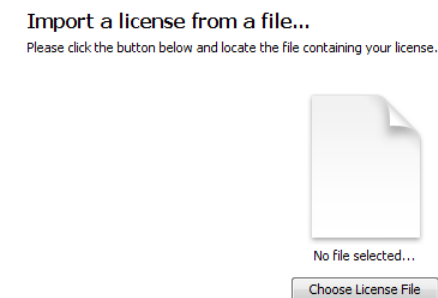


Figure 1.10: Importing the license file downloaded from the web page.

Accepting the license agreement

The end user license agreement (EULA) must be read and accepted as part of the installation process. figure 1.28.

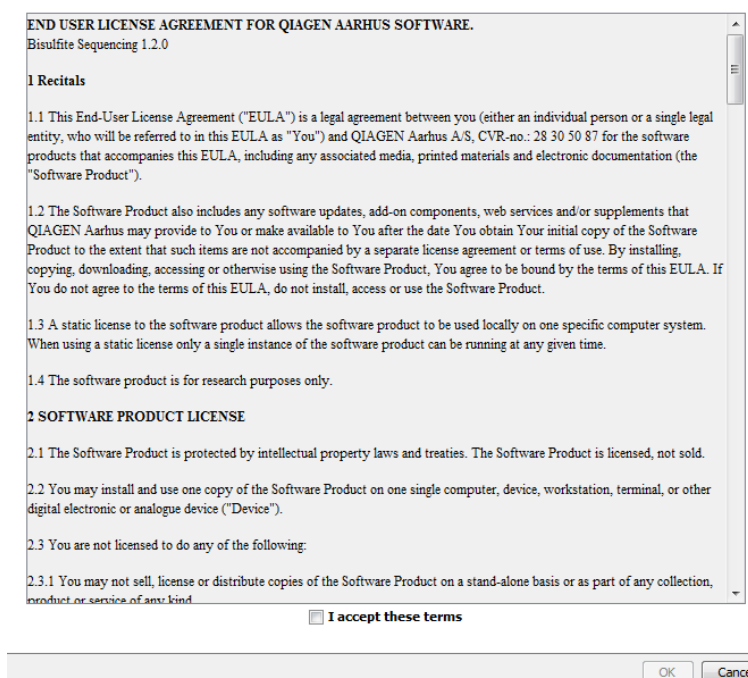


Figure 1.11: The End User License Agreement is presented during the installation process.

Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

1.4.3 Import a license from a file

If you already have a license file associated with the host ID of your machine, it can be imported using this option.

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

Click on the **Choose License File** button, locate the license file and selected it. Then click on the **Next** button.

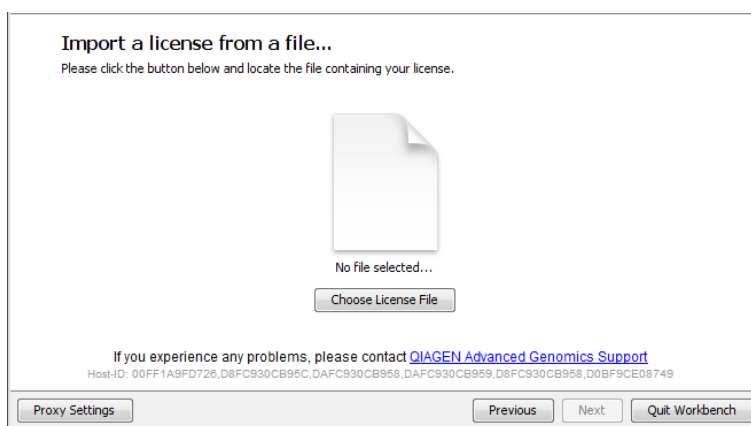


Figure 1.12: Selecting a license file.

Accepting the license agreement

The end user license agreement (EULA) must be read and accepted as part of the installation process. figure 1.28.

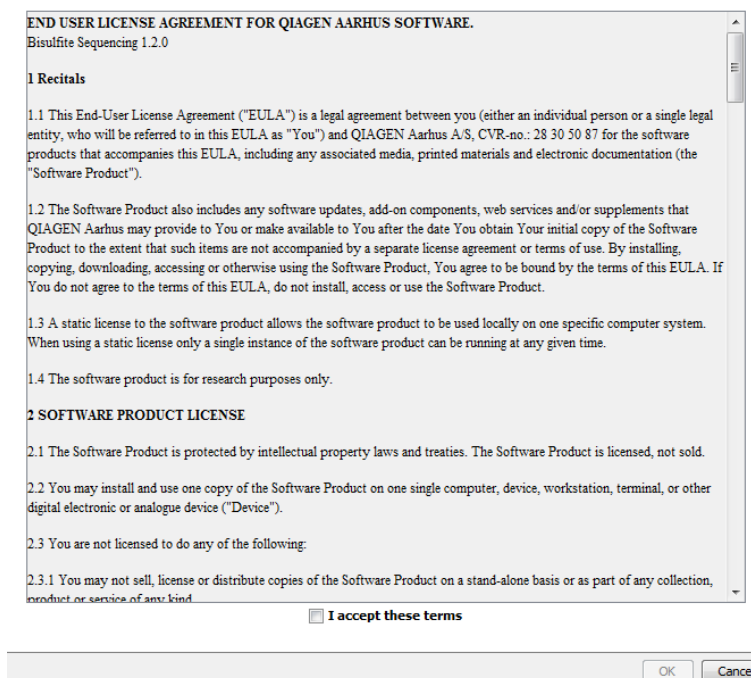


Figure 1.13: The End User License Agreement is presented during the installation process.

Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

1.4.4 Upgrade license

The option "Upgrade from an existing Workbench installation" can be convenient when you have been using another version of a licensed Workbench and the license is covered by our Maintenance, Upgrades and Support (MUS) program. Licenses not covered by MUS cannot be updated to support a new major Workbench release line.

If your license is covered our Maintenance, Upgrades and Support (MUS) program but you experience problems downloading a license for the new version of the software, please contact bioinformaticslicense@qiagen.com.

The Workbench will need direct access to the external network to use this option. If the Workbench cannot connect to the external network directly, please see section 1.4.6.

After selecting the "Upgrade from an existing Workbench installation" option, click on the **Next** button. The Workbench will search for an earlier installation of the same Workbench product you are upgrading to.

If it finds that installation, it will locate the existing license file and show information like that in figure 1.14.

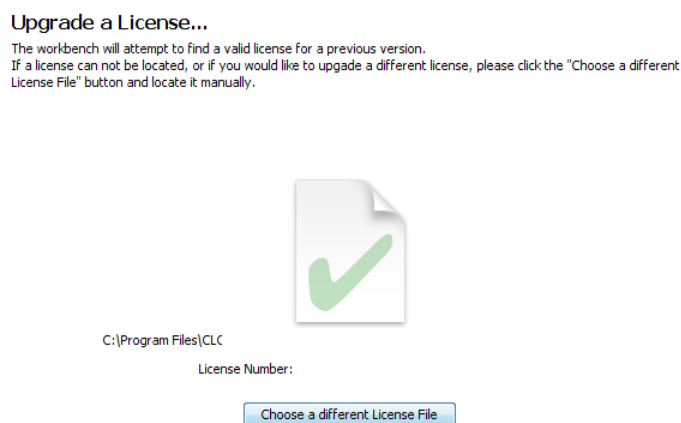


Figure 1.14: An license from an older installation was found.

When you click on the **Next** button, the Workbench checks if you are entitled to upgrade your license. This is done by contacting QIAGEN Aarhus servers.

If the earlier Workbench version could not be found, which can be the case if you have installed to a custom location or are upgrading from one Workbench product to another product replacing it¹, then click on the "Choose a different License File" button. Navigate to where the older license file is, which will be in a subfolder called "licenses" within the installation area of the Workbench you are upgrading from. Select the license file and click on the "Open" button.

If the license selected can be updated, a message similar to that shown in figure 1.15 will be displayed. If there is a problem updating the selected license, a dialog will appear indicating this.

Click on the **Next** button and then choose how to proceed to get the updated license file.

In this dialog, there are two options:

- **Direct Download.** Download the license directly. This method requires that the Workbench has access to the external network.
- **Go to CLC License Download web page.** The online license download form will be opened in a web browser. This option is suitable for when downloading a license for use on another

¹In November 2018, the Biomedical Genomics Workbench was replaced by the CLC Genomics Workbench and a free plugin, Biomedical Genomics Analysis. Licenses for the Biomedical Genomics Workbench covered by MUS at that time can be used to download a valid license for the CLC Genomics Workbench, but the upgrade functionality is not able to automatically find the older license file.

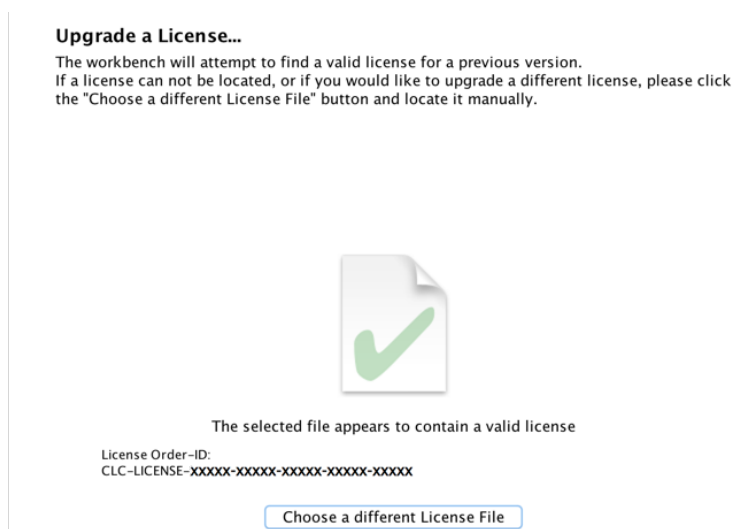


Figure 1.15: An license from an older installation was found.

machine that does not have access to the external network, and thus cannot access the QIAGEN Aarhus servers.

After selecting your method of choice, click on the button labeled **Next**.

Direct download

After choosing the **Direct Download** option and clicking on the button labeled **Next**, a dialog similar to that shown in figure 1.16 will appear if the license is successfully downloaded and installed.

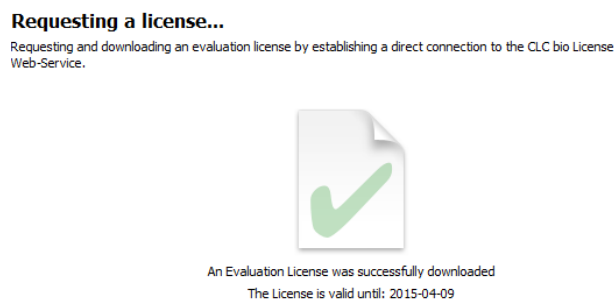


Figure 1.16: A license has been successfully downloaded and installed for use.

When the license has been downloaded and installed, the **Next** button will be enabled.

If there is a problem, a dialog will appear indicating this.

Go to license download web page

After choosing the **Go to CLC License Download web page** option and clicking on the button labeled **Next**, the license download form will be opened in a web browser, as shown in figure 1.17.

Click on the **Download License** button and then save the license file.

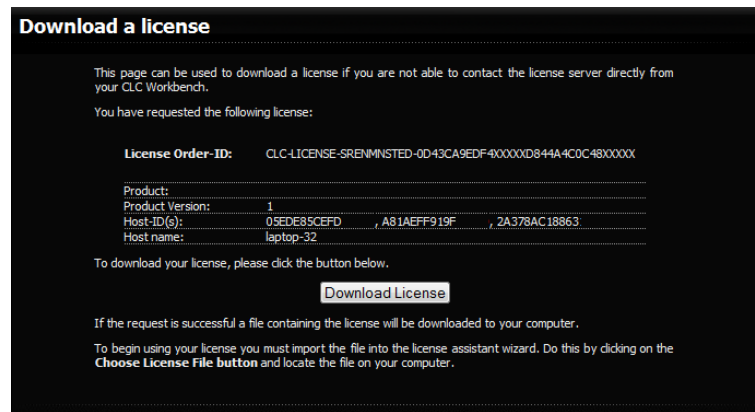


Figure 1.17: The license download form opened in a web browser.

Back in the Workbench window, you will now see the dialog shown in 1.18.

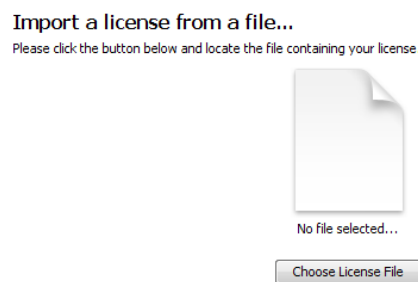


Figure 1.18: Importing the license file downloaded from the web page.

Click on the **Choose License File** button, find the saved license file and select it. Then click on the **Next** button.

Accepting the license agreement

The end user license agreement (EULA) must be read and accepted as part of the installation process. figure 1.28.

Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

1.4.5 Configure license server connection

If your organization is running a CLC License Server, you can configure your Workbench to connect to it to get a license.

To configure the Workbench to connect to a CLC License Server, select the **Configure License Server connection** option and click on the **Next** button. A dialog for the license server connection configuration is then presented, as shown in figure 1.20.

The options in that dialog are:

- **Enable license server connection.** This box must be checked for the Workbench is to contact the CLC License Server to get a license for *CLC Main Workbench*.

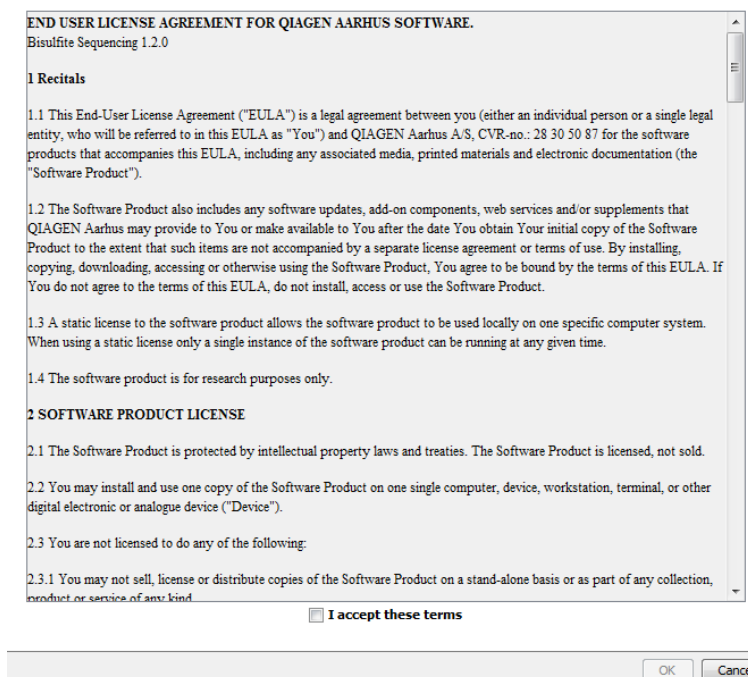


Figure 1.19: The End User License Agreement is presented during the installation process.

Configure License Server connection...

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

Enable license server connection

Automatically detect license server.

Manually specify license server:

Hostname/IP-address:

Port:

Use custom username when requesting a license

Username:

Disable license borrowing

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

Figure 1.20: Connecting to a CLC License Server.

- **Automatically detect license server.** By checking this option the Workbench will look for a CLC License Server accessible from the Workbench. Automatic server discovery sends UDP broadcasts from the Workbench on port 6200. Available license servers respond to the broadcast. The Workbench then uses TCP communication for to get a license, if one is available. Automatic server discovery works only on local networks and will not work on WAN or VPN connections. Automatic server discovery is not guaranteed to work on all networks. If you are working on an enterprise network on where local firewalls or routers cut off UDP broadcast traffic, then you may need to configure the details of the CLC License Server using the **Manually specify license server** option instead.
- **Manually specify license server.** Select this option to enter the details of the machine the CLC License Server software is running on, specifically:
 - **Host name.** The address of the machine the CLC Licenser Server software is running

on.

- **Port.** The port used by the CLC License Server to receive requests.
- **Use custom username when requesting a license.** Optional. If this is checked, a username can be entered. That will be passed to the CLC License Server instead of the username of the account being used to run the Workbench.
- **Disable license borrowing on this computer.** Check this box if you do not want users of the computer to borrow a license. See section [1.4.5](#) for further details.

Special note on modules needing a license

This note concerns CLC Genomics Workbench 11.0 and higher, Biomedical Genomics Workbench 5.0 and higher and CLC Main Workbench 8.0 and higher.

A valid module license is needed to start a module tool, or a workflow including a module tool. Module licenses obtained through a License Server connection will be valid for four hours after starting the tool or the workflow. A process started (whether a module tool or a workflow including a module tool) will always be completed, even if its completion exceeds the four hours period where the license is valid.

If the tool or the workflow completes before the four hour validity period, it is possible to start a new tool or a workflow, and this will always refresh the validity of the license to a full four hours period. However, if the tool or the workflow completes after the four hour validity period, a new license will need to be requested after that to start the next tool or workflow.

These measures ensure that more licenses are available to active users, rather than blocked on an inactive computer, i.e., where the workbench would be open but not in use.

Borrowing a license

A *CLC Main Workbench* using a network license normally needs to maintain a connection to the CLC License Server. However, if allowed by the network license administrator, network licenses can be *borrowed* for offline use. During the period a license has been borrowed, there will be one less network license available for other users.

If CLC License Server administrator has chosen not to allow the borrowing of network licenses, then the information in this section is not relevant.

The Workbench must be connected to the CLC License Server at the point when the license is borrowed. The procedure for borrowing a license is:

1. Go to the Workbench menu option:
Help | License Manager
2. Click on the "Borrow License" tab to display the dialog shown in figure [1.21](#).
3. Select the license(s) that you wish to borrow by clicking in the checkboxes in the Borrow column in the License overview panel.
4. Choose the length of time you wish to borrow the license(s) for using the drop down list in the Borrow License tab. By default the maximum is 7 days, but network license administrators can specify a lower limit than this.

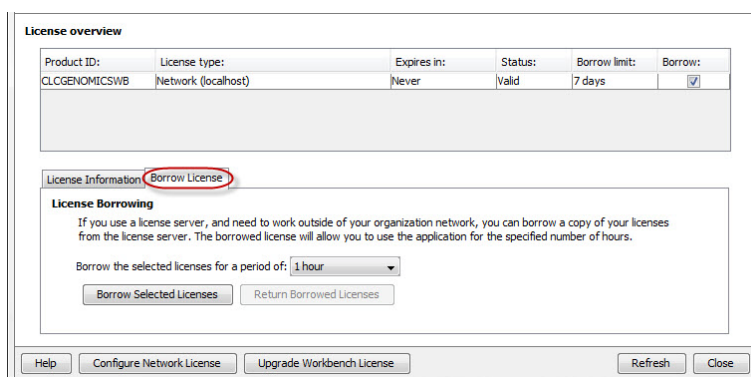


Figure 1.21: Borrow a license.

5. Click on the button labeled **Borrow Selected Licenses**.
6. Close the License Manager when you are done.

You can now go offline and continue working with the *CLC Main Workbench*. When the time period you borrowed the license for has elapsed, the network license will be again made available for other users. To continue using *CLC Main Workbench* with a license, you will need to connect to the network again so the Workbench can contact the CLC Licence Server.

You can return borrowed licenses early if you wish by started up the **License Manager**, opening the "Borrow License" tab, and clicking on the **Return Borrowed Licenses** button.

Common issues when using a network license

- No license available at the moment If all licenses are in use, you will see a dialog like that shown in figure 1.22 when you start up the Workbench.

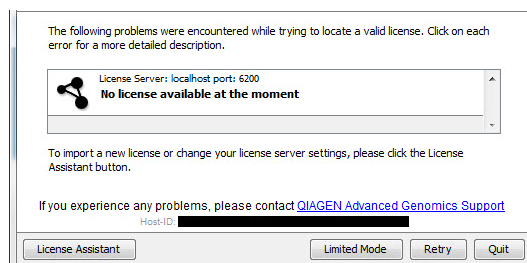


Figure 1.22: This window appears when there are no available network licenses for the software you are running.

You will need to wait for at least one license to be returned before you can continue to work with a fully functional copy of the software. If running out of licenses is a frequent issue, you may wish to discuss this with your CLC License Server administrator.

Clicking on the **Viewing Mode** button in the dialog allows you to run the *CLC Main Workbench* for viewing data, and for basic analyses, import and export. Please see section 1.4.7 for further details.

- Lost connection to the CLC License Server If the Workbench connection to the CLC License Server is lost, you will see a dialog like that shown in figure 1.23.

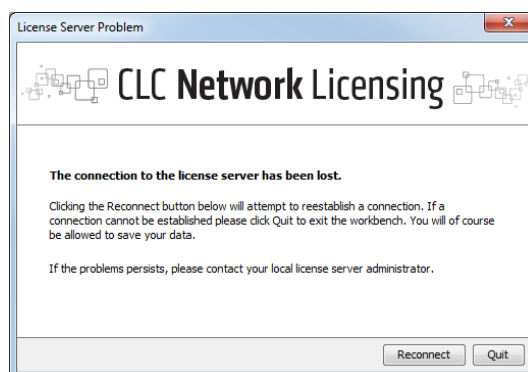


Figure 1.23: Here, the Workbench is unable to establish a connection to a CLC License server.

If you have chosen the option to **Automatically detect license server** and you have not succeeded in connecting to the CLC License Server before, please check with your local IT support that automatic detection will be possible to do at your site. If it is not, you will need to specify the CLC License Server settings, as described earlier in this section.

If you have successfully contacted the CLC License Server from your Workbench previously, please consider discussing this issue with your CLC License Server administrator or your local IT support, for example, making sure that the CLC License Server is running and that your Workbench is able to connect to it.

There may be situations where you wish to use a different license or view information about the license(s) the Workbench is currently using. To do this, open the License Manager using the menu option:

Help | License Manager ()

The license manager is shown in figure 1.24.

This License Manager can be used to:

- See information about the license (e.g. the license type, when it expires, etc.)
- Configure the connect to a CLC License Server. Click on the **Configure Network License** button at the lower left corner to open the dialog seen in figure 1.20.
- Upgrade from an evaluation license. Click on the **Upgrade Workbench License** button to open the dialog shown in figure 1.1.
- Export license information to a text file.
- Borrow a license, relevant when using a network license.

If you wish to switch away from using a network license, click on the button to **Configure Network License** and uncheck the box beside the text **Enable license server connection** in the dialog. When you restart the Workbench, you can set up the new license as described in section 1.4.

1.4.6 Download a static license on a non-networked machine

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

License overview

Product ID:	License type:	Expires in:	Status:	Borrow limit:	Borrow:
METAGENEMARK	Network (10.1.10.1)	Never	Valid	7 days	<input checked="" type="checkbox"/>
CLCGENOMICSWB	Network (10.1.10.1)	Never	Valid	7 days	<input checked="" type="checkbox"/>
CLC_MICROBIAL_GE...	Network (10.1.10.1)	Never	Valid	7 days	<input checked="" type="checkbox"/>
CLC_MICROBIAL_GE...	Local Evaluation License	11 Days	Valid	-	

License Information | Borrow License

License Information

Product ID:

Order ID:

Licensee:

Expires:

Source:

Local Machine Information

Hostname: laptop-112

Host ID: 00FF1A9FD726,D8FC930CB95C,DAFC930CB958,DAFC930CB959,D8FC930CB958,D08F9CE08749

Export License Information

Help | Configure Network License | Upgrade Workbench License | Refresh | Close

Figure 1.24: The License Manager provides information about licenses being used and access to other license-related functionality.

- 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 of the machine at the bottom of the License Assistant window in grey text, or, if working in Viewing Mode, by launching the **License Manager** from under the Workbench **Help** menu option.
- 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 network license download web page²:
<https://secure.clcbio.com/LmxWSv3/GetLicenseFile>
- Paste in your license order ID and the host ID that you noted down in the relevant boxes on the web page.
- Click on '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 on the 'Choose License File' button and then locate and select 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 menu option:

Help | License Manager (📄)

- Click on the **Next** button and go through the remaining steps to install the license.

²For CLC Genomics Workbench 5.x and earlier or CLC Main Workbench 6.7.x and earlier, the license download page URL is <http://licensing.clcbio.com/LmxWSv1/GetLicenseFile>

1.4.7 Viewing mode

Using a CLC Workbench in Viewing Mode is a free and easy way to access extensive data viewing capabilities, basic bioinformatics analysis tools, as well as import and export functionality.

Data viewing

Any data type supported by the Workbench being used can be viewed in Viewing Mode. Plugins or modules can also be installed when in Viewing Mode, expanding the range of data types supported.

Viewing Mode of the CLC Workbenches can be particularly useful when sharing data with colleagues or reviewers who wish to view and investigate data you have generated but who do not have access to a Workbench license.

Data import, export and analysis in Viewing Mode

When working in Viewing Mode, the Import and Export buttons in the top Toolbar are enabled, and standard import and export functionality for many bioinformatics data types is supported. Tools available can be seen in the Workbench Toolbox, as illustrated in figure 1.25.

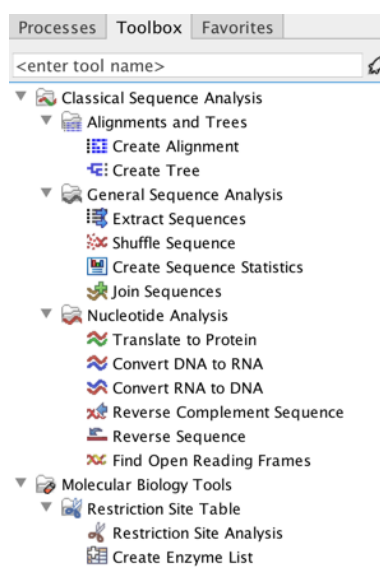


Figure 1.25: Bioinformatics tools available when using Viewing Mode are found in the Toolbox.

Starting a CLC Workbench in Viewing Mode

A button labeled **Viewing Mode** is presented in the Workbench License Manager when a Workbench is started up without a license installed, as shown in figure 1.26. This button is also visible in message windows that appear if a Workbench is started up that has an expired license or that is configured to use a network license but all the available licenses have been checked out by others, as described in section 1.4.5.

Click on the **Viewing Mode** button to start up the Workbench in Viewing Mode.

To go from running in Viewing Mode to running a Workbench with its full functionality, it just needs to have access to a valid license. This can be done by installing a static license, or when using a network license, by restarting the Workbench when licenses are once again available.

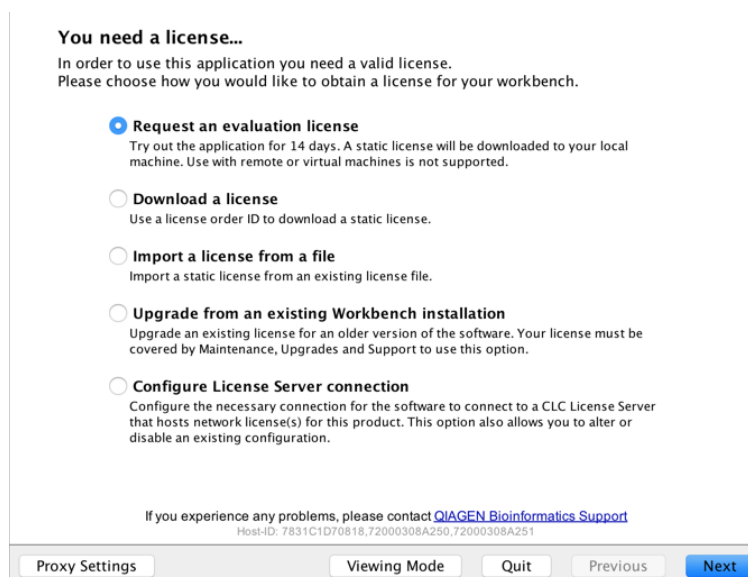


Figure 1.26: Click on the *Viewing Mode* button at the bottom of the *License Manager* window to launch the *Workbench* in *Viewing Mode*.

1.4.8 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 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.qiagenbioinformatics.com/plugins/> for a full list of plugins with descriptions of their functionalities.

Note: In order to install plugins and modules, the Workbench must be run in administrator mode. On Linux and Mac, it means you must be logged in as an administrator. On Windows, you can do this by right-clicking the program shortcut and choosing "Run as Administrator".

Plugins are installed and uninstalled using the plugin manager.

Help in the Menu Bar | **Plugins...** () or **Plugins** () **in the Toolbar**

The plugin manager has two 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 QIAGEN Aarhus server.

1.5.1 Install

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 1.27).

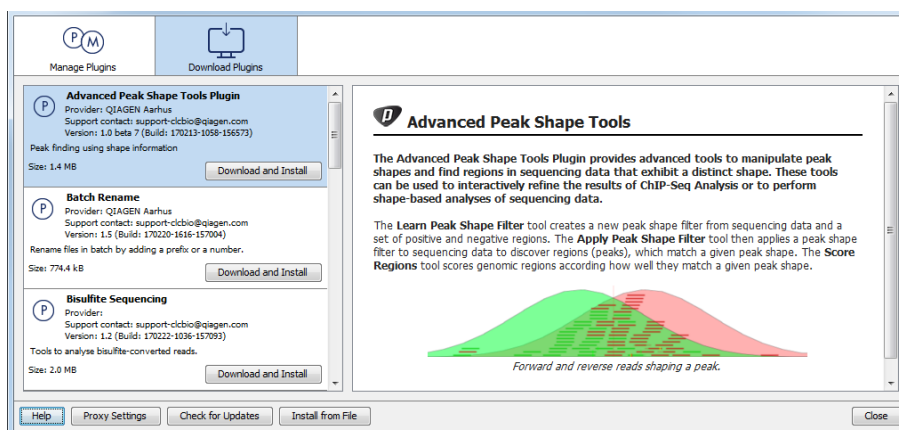


Figure 1.27: The plugins that are available for download.

Select the plugin of interest to display additional information about the plugin on the right side of the dialog. Click **Download and Install** to add the plugin functionalities to your workbench.

Accepting the license agreement

The end user license agreement (EULA) must be read and accepted as part of the installation process. figure 1.28.

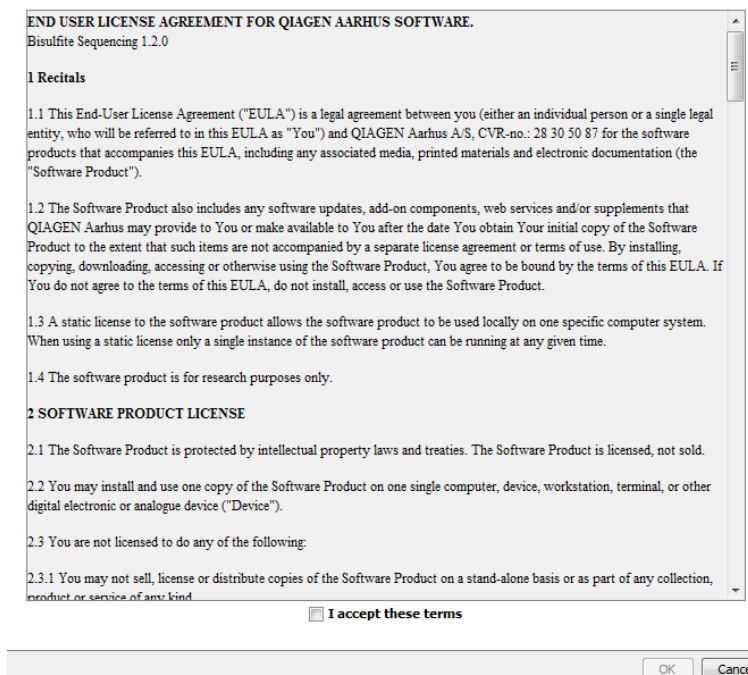


Figure 1.28: The End User License Agreement is presented during the installation process.



Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

If the plugin is not shown on the server but you have the installer file on your computer (for example if you have downloaded it from our website), you can install the plugin by clicking the **Install from File** button at the bottom of the dialog and specifying the plugin *.cpa file saved on your computer.

When you close the dialog, you will be asked whether you wish to restart the workbench. The plugin will not be ready for use until you have restarted.

1.5.2 Uninstall

Plugins are uninstalled using the plugin manager:

Help in the Menu Bar | Plugins... () or Plugins () in the Toolbar

This will open the dialog shown in figure 1.29.

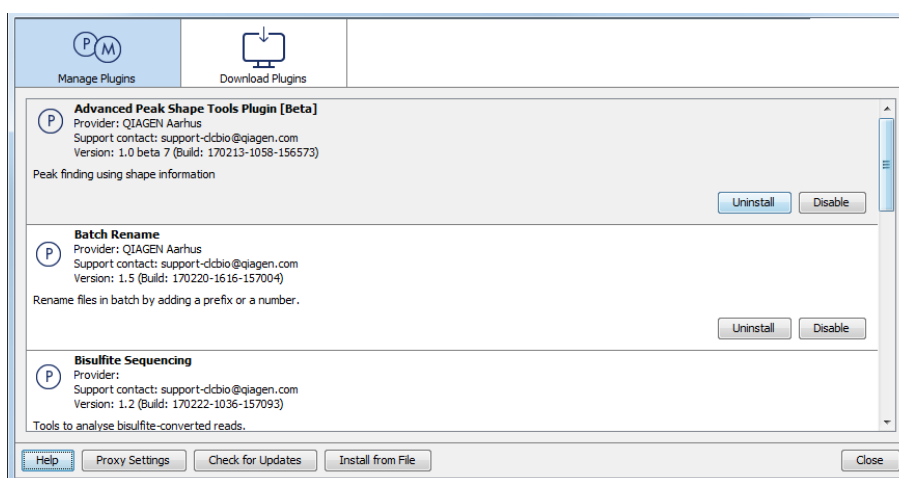


Figure 1.29: The plugin manager with plugins installed.

The installed plugins are shown in the **Manage plugins** tab of the plugin manager. To uninstall, select the plugin in the list and click **Uninstall**.

If you do not wish to completely uninstall the plugin, but you do not want it to be used next time you start the Workbench, click the **Disable** button.

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

1.5.3 Updating plugins

If a new version of a plugin is available, you will get a notification during start-up as shown in figure 1.30.

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 1.29).

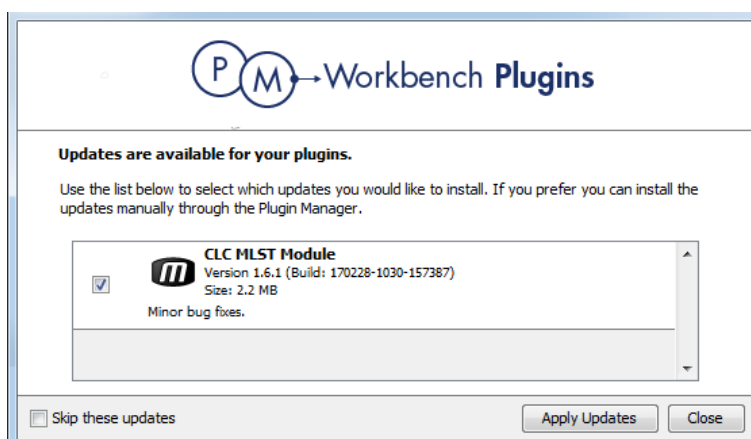


Figure 1.30: Plugin updates.

1.6 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.

CLC Main Workbench supports the use of an HTTP-proxy and an anonymous SOCKS-proxy.

To configure your proxy settings, open the workbench, go to **Edit | Preferences** and choose the **Advanced** tab (figure 1.31).

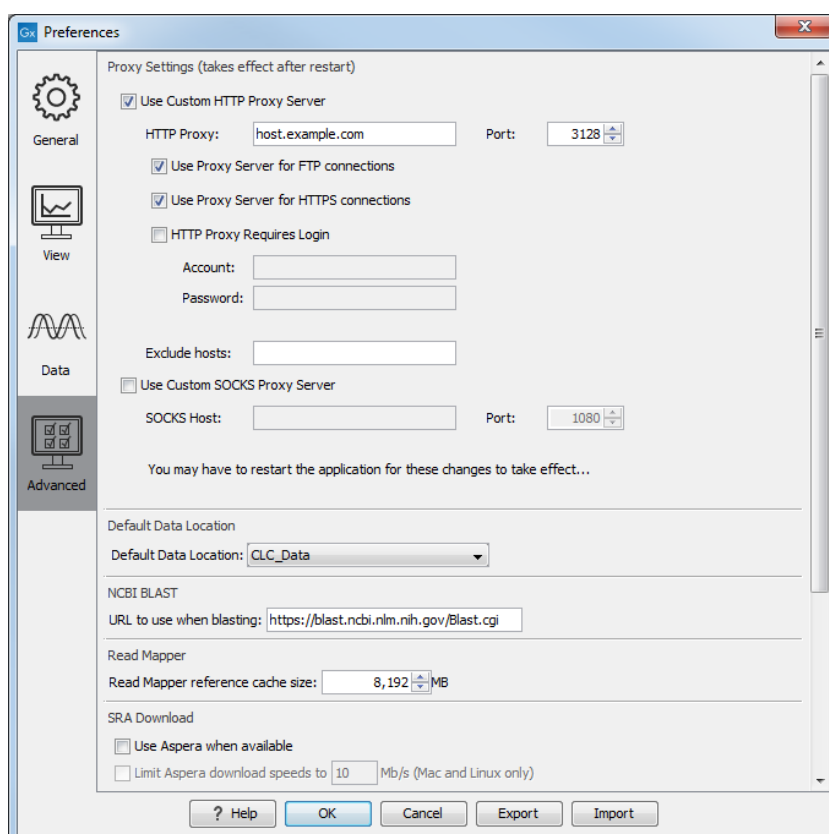


Figure 1.31: Adjusting proxy preferences.

You have the choice between an HTTP-proxy and a SOCKS-proxy. The 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 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.

1.7 CLC Server connection

Using a *CLC Server*, data can be stored centrally and analyses run on a central machine rather than on a personal computer. After logging into the *CLC Server* from a Workbench, data in *CLC Server* locations will be listed in the Workbench Navigation Area. When launching analyses that can be run on the *CLC Server*, you will be offered the choice of running them using the Workbench or the *CLC Server*.

1.7.1 CLC Server Connection

To log into a *CLC Server* or to check on the status of an existing connection, go to:

File | CLC Server Connection (S)

This will bring up a login dialog as shown in figure 1.32.

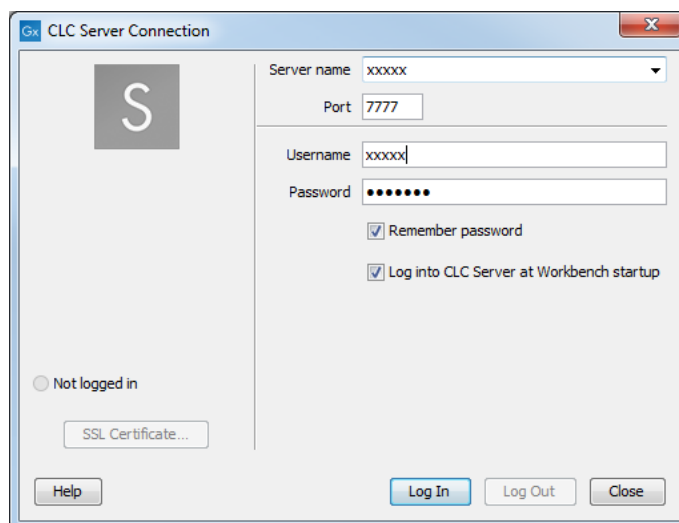


Figure 1.32: The CLC Server Connection dialog.

To log into a *CLC Server*, fill in the text fields. Your server administrator should be able to provide you with the necessary details. When you click on the **Log In** button, the Workbench will connect to the *CLC Server* if your credentials are accepted.

Your username and the server details will be saved between Workbench sessions. If you wish your password to be saved also, click in the box beside the **Remember password** box.

If you wish the Workbench to connect to the server automatically on startup, then check the box beside the option **Log into CLC Server at Workbench startup**. This option is only enabled when the **Remember password** option has been selected.

Further information about working with a *CLC Server* from a *CLC Workbench* is available in this manual:

- Launching tasks on a *CLC Server* is described in section [8.2.1](#).
- Monitoring processes sent to the *CLC Server* from a *CLC Workbench* is described in section [2.3](#).
- Viewing and working with data held on a *CLC Server* is described in section [3.1.1](#), and deleting data held on a *CLC Server* is described in section [3.1.7](#).
- Importing data to a *CLC Server* and exporting data held on a *CLC Server* is described in section [6.5](#).

For those logging into the *CLC Server* as a user with administrative privileges, an option called *Manage Server Users and Groups...* will be available. This is described in detail at http://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=User_authentication_using_Workbench.html.

1.8 Getting started and latest improvements

CLC Main Workbench includes an extensive **Help** function, which can be found in the **Help** menu of the program's **Menu bar** (or by pressing F1).

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.qiagenbioinformatics.com/support/tutorials/>. We also recommend our **Online presentations** where a product specialist 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://tv.qiagenbioinformatics.com/>.

Finally, *CLC Main Workbench* is being constantly developed and improved. A detailed list of new features, improvements, bug fixes, and changes for the current version of *CLC Main Workbench* can be found at <http://www.qiagenbioinformatics.com/products/latest-improvements/>.

History of the CLC Workbenches

In November 2005, CLC bio releases 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 with a range of more advanced analyses.

In March 2006, *CLC DNA Workbench* (formerly *CLC Gene Workbench*) and *CLC Main Workbench* are 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 has additional advanced features. *CLC Main Workbench* holds all basic and advanced features of the *CLC Workbenches*.

In June 2007, *CLC RNA Workbench* is 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, *CLC Free Workbench* changes name to *CLC Sequence Viewer*.

In June 2008, the first version of the *CLC Genomics Workbench* is 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.

In December 2006, CLC bio releases 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* are discontinued. All customers with a valid license for any of these products are offered an upgrade to the *CLC Main Workbench*.

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

In April 2014, CLC bio releases *CLC Cancer Research Workbench*, a product that contains streamlined data analysis workflows with integrated trimming and quality control tailored to meet the requirements of clinicians and researchers working within the cancer field.

In April 2015, *CLC Cancer Research Workbench* is renamed to *Biomedical Genomics Workbench* to reflect the inclusion of tools addressing the requirements of clinicians and researchers working within the hereditary disease field in addition to the tools designed for those working within the cancer field.

In June 2017, Viewing Mode is introduced in all commercial CLC Workbenches. This mode is available when a Workbench is launched without a valid license. In this mode, data can be viewed and some basic analyses equivalent to those available in the free CLC Sequence Viewer, can be run.

In January 2018, **CLC Drug Discovery Workbench** is discontinued.

In November 2018, the *Biomedical Genomics Analysis* plugin is released for use with *CLC Genomics Workbench*. With the Biomedical Genomics Analysis plugin installed, CLC Genomics Workbench becomes the delivery mechanism for all biomedical analyses previously delivered by *Biomedical Genomics Workbench* and some associated plugins. *Biomedical Genomics Workbench* is correspondingly discontinued, and all customers with valid licenses for *Biomedical Genomics Workbench* can use them for *CLC Genomics Workbench*.

Part II

Core Functionalities

Chapter 2

User interface

Contents

2.1	View Area	41
2.1.1	Open view	41
2.1.2	History and Elements Info views	43
2.1.3	Close views	44
2.1.4	Save changes in a view	45
2.1.5	Undo/Redo	46
2.1.6	Arrange views in View Area	46
2.1.7	Moving a view to a different screen	47
2.1.8	Side Panel	48
2.2	Zoom and selection in View Area	49
2.2.1	Zoom in	50
2.2.2	Zoom out	50
2.2.3	Selecting, panning and zooming	51
2.3	Toolbox and Status Bar	51
2.3.1	Processes	52
2.3.2	Toolbox	53
2.3.3	Favorites	53
2.3.4	Status Bar	54
2.4	Workspace	55
2.5	List of shortcuts	56

This chapter provides an overview of the different areas in the user interface of *CLC Main Workbench*. As can be seen from figure 2.1 this includes:

- a **Navigation Area** where files are sorted;
- a **Toolbox** that can be opened as such, or as a Processes or a Favorites tab;
- a **View Area** with one or more tabs open;
- a **Side Panel** where it is possible to change the settings for the currently opened View;

- a **Menu Bar** to access various function, and a **Toolbar** that highlights the most common actions;
- a **Status Bar** at the bottom of the screen that indicates the status of the workbench (processing a job, or idle) and additional information that are View-dependant.

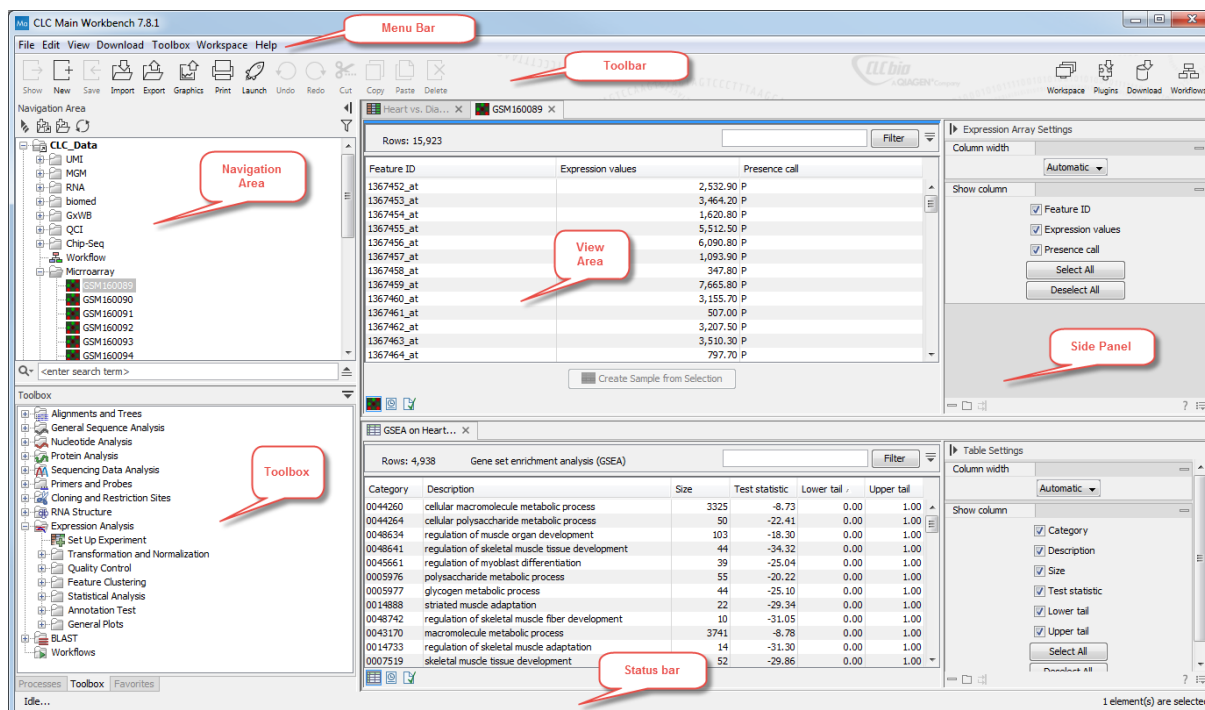


Figure 2.1: The user interface.

2.1 View Area

The **View Area** is the central 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. In figure 2.2, four views are displayed: three as tabs in the upper view, and one in an horizontal split view. The tab currently selected, i.e., active, is indicated by a blue bar underneath the tab (here the bottom tab open in the bottom view).

Switch tabs in View Area using the following shortcuts Ctrl + PageUp or PageDown (or ⌘ + PageUp or PageDown on Mac).

Several operations can be performed by right-click menus that can be activated from the tab, or by using the icon list at the bottom of each view.

2.1.1 Open view

Elements

Opening an element can be done in a number of ways:

double-click an element in the Navigation Area

or **select an element in the Navigation Area | Show** or **Ctrl + O** (⌘ + B on Mac)

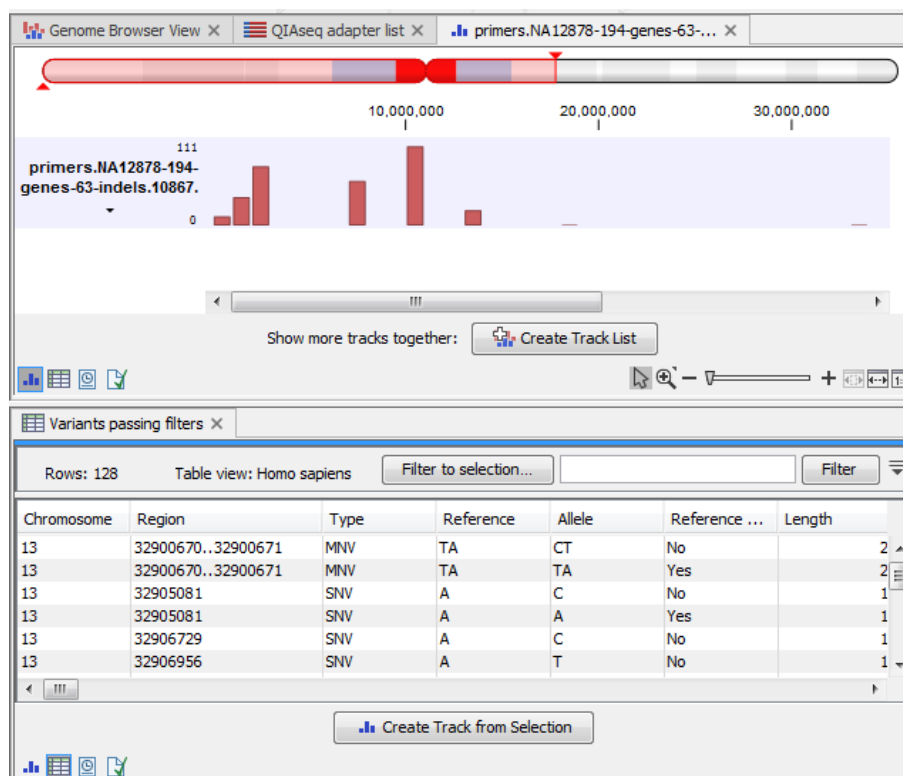


Figure 2.2: A View Area can enclose several views, each view indicated with a tab.

Opening an element while another element is already open in the View Area will show the new element in front of the other. The element that was already open can be brought to front by clicking its tab.

Views

Each element can be shown in different ways. A sequence, for example, can be shown as linear, circular, text, etc.

For example, to see a linear sequence in a circular view, open the sequence as linear in the View Area and

Click Show As Circular (○) at the lower left part of the view

The buttons used for switching views are shown in figure 2.3. They are element-dependent, meaning that different elements may have different buttons available. You can switch from one to the other sequentially by clicking Ctrl + Shift + PageUp or Ctrl + Shift + PageDown.



Figure 2.3: The buttons shown at the bottom of a view of a nucleotide sequence. You can click the buttons to change the view to a circular view or a history view.

Split views

If the sequence is already open in a linear view (ACT), and you wish to see both a circular and a linear view, you can split the views very easily:

Press Ctrl (⌘ on Mac) while you | Click Show As Circular (○) at the lower left part of the view

This will open a split view with a linear view at the bottom and a circular view at the top (see 11.5).

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 (⊙)

2.1.2 History and Elements Info views

The two buttons to the right hand side of the toolbar are **Show History** (📄) and **Show Element Info** (📄).

The History view is a textual log of all operations you make in the program. If for example 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.

When an element's history is opened, the newest change is submitted in the top of the view (figure 2.4).

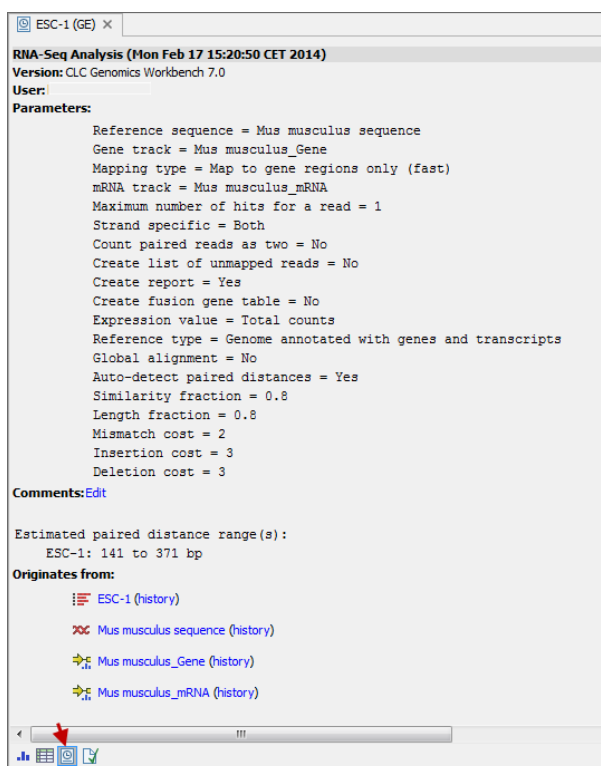



Figure 2.4: An element's history.

The following information is available:

- **Originates from workflow** (optional). In cases where the file was generated by a workflow, the first line will state the Name and Version number of that workflow.
- **Title**. The action that the user performed.
- **Date and time**. Date and time for the operation. The date and time are displayed according to your locale settings (see section 4.1).

- **Version.** The workbench type and version that has been used.
- **User.** The user who performed the operation. If you import some data created by another person in a CLC Workbench, that person's name will be shown.
- **Parameters.** Details about the action performed. This could be the parameters that were chosen for an analysis.
- **Comments.** By clicking **Edit** you can enter your own comments regarding this entry in the history. These comments are saved.
- **Originates from.** This information is usually shown at the bottom of an element's history. Here, you can see which elements the current element originates from. For example, if you have 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.

When an element's info is open you can check current information about the element, and in particular the potential association of the data you are looking at with metadata. To learn more about the **Show Element Info** () button, see section 11.4 and see section 3.2.4.

2.1.3 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 | Close or **Select the view | Ctrl + W**

By right-clicking a tab, the following close options exist (figure 2.5).

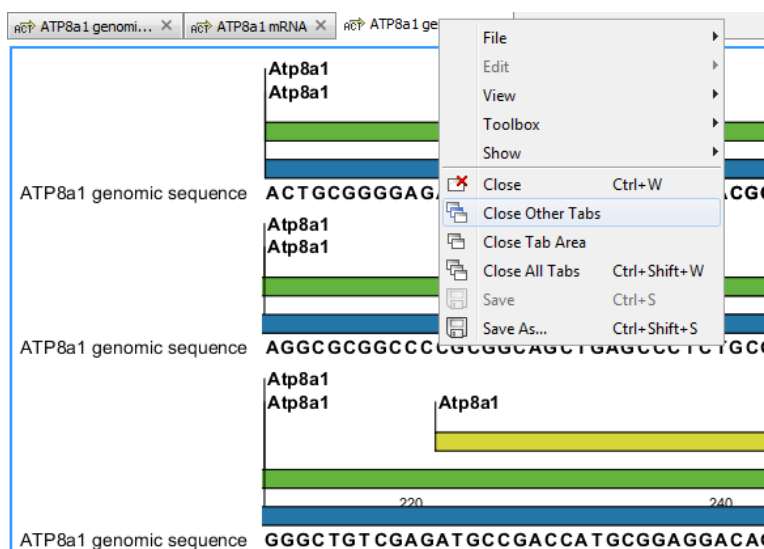


Figure 2.5: By right-clicking a tab, several close options are available.

- **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, but not the tabs that are in split view.
- **Close All Tabs.** Closes all tabs, in all tab areas. Leaves an empty workspace.

2.1.4 Save changes in a view

When a new view is created, an * in front of the name of the view in the tab indicates that the element has not been saved yet. Similarly, when changes to an element are made in a view, an * is added before the element name on the tab and the element name is shown in *bold and italic* in the Navigation Area (figure 2.6).

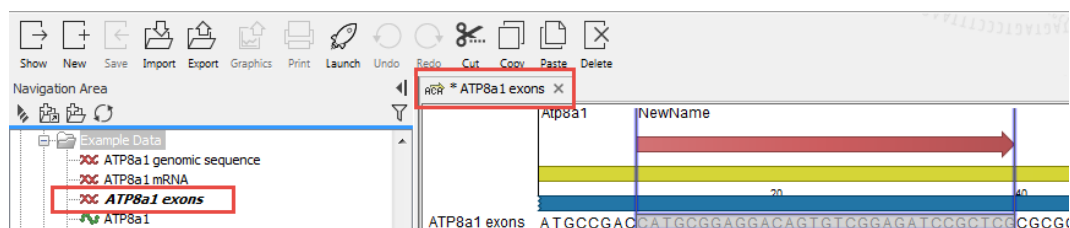


Figure 2.6: An * on a tab name always indicates that the view is unsaved. In this case, an existing element was edited but not saved yet, so the element's name is also highlighted in bold and italic in the Navigation Area.

The **Save** function may be activated in two ways: Select the tab of the view you want to save and **Save** (⌘) or **Ctrl + S** (⌘ + S on Mac)

If you close a tab of a view containing an element that was edited, you will be asked if you want to save.

When saving an element from a new view that has not been opened from the Navigation Area, a save dialog appears (figure 2.7). In this dialog, you can name the element and select the folder in which you want to save the element.

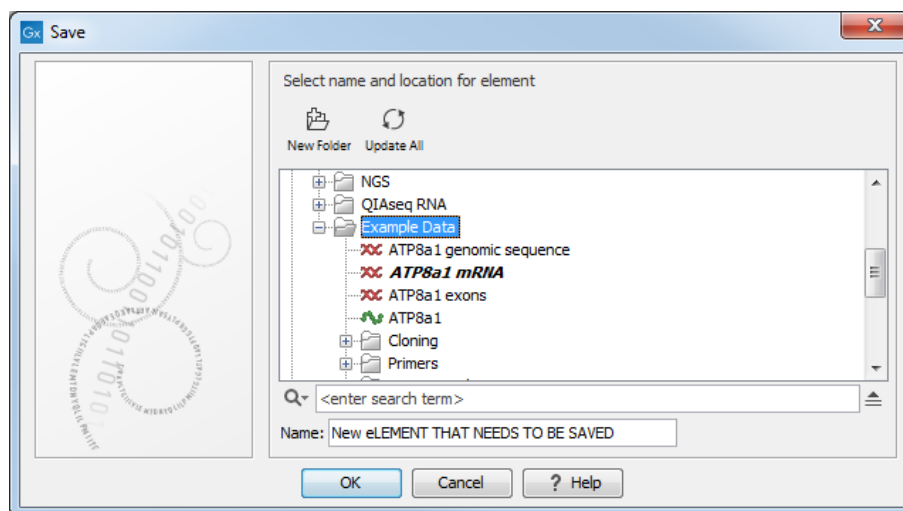


Figure 2.7: Save dialog. The new element has been name "New element that needs to be saved" and will be saved in the "Example Data" folder.

2.1.5 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 **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 **Ctrl + Y**

Note! Actions in the Navigation Area, e.g., renaming and moving elements, cannot be undone. However, you can restore deleted elements (see section 3.1.7).

You can set the number of possible undo actions in the Preferences dialog (see section 4).

2.1.6 Arrange views in View Area

To provide more space for viewing data, you can hide Navigation Area and Toolbox by clicking the hide icon (⏏) at the top of the Navigation Area. You can also hide the Side Panel using the same icon at the top of the Side Panel.

Views are arranged in the **View Area** by their tabs. The order of the views can be changed using drag and drop.

If a tab is dragged into a view, the area where the tab will be placed is highlighted blue. The blue area can be a tab bar in another view, or the bottom of an existing view. In that case, the tab will be moved to a new split view.

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. When the split is made vertically, the new view opens to the right of the existing view (see figure 2.8).

Splitting the View Area can be undone by dragging the tab of the bottom view to the tab of the top view, or by using the **Maximize/Restore View** function.

Select the view you want to maximize, and click

View | Maximize/restore View (☐) or Ctrl + M

or **right-click the tab | View | Maximize/restore View (☐)**

or **double-click the tab of view**

The following restores the size of the view:

View | Maximize/restore View (☐) or Ctrl + M

or **double-click title of view**



Figure 2.8: A vertical split screen.

2.1.7 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 2.9, 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.

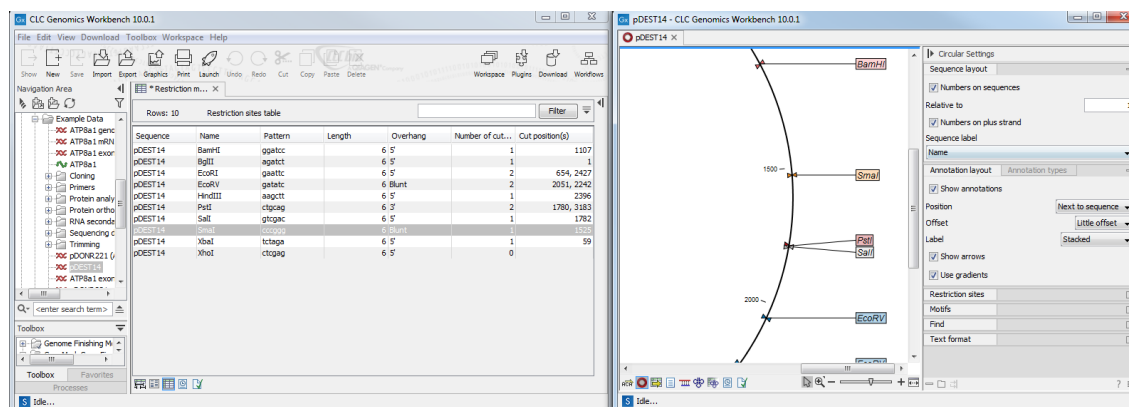


Figure 2.9: Showing the table on one screen while the sequence is displayed on another screen. Clicking the table of open reading frames causes the view on the other screen to follow the selection.

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.

2.1.8 Side Panel

The **Side Panel** allows you to change the way the content of a view is 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 2.10 shows the default Side Panel for a protein sequence. It is organized into **palettes**.

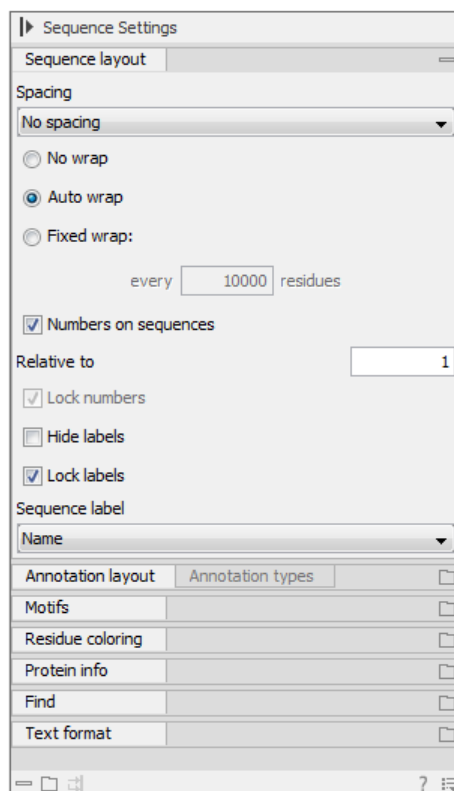


Figure 2.10: The default view of the Side Panel when opening a protein sequence.

In this example, there is one palette 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 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 2.11.

In this example, the Motifs palette has been placed on top of the sequence view together with the the Residue coloring palette. 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 2.12) there are a number of icons used to:

- Collapse all settings (=).

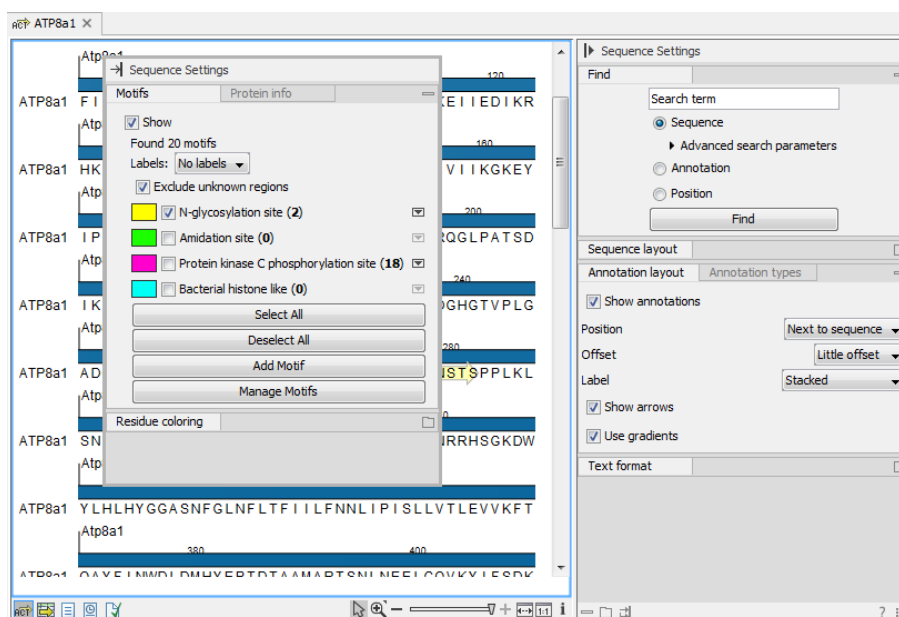


Figure 2.11: Palettes can be organized in the Side Panel as you like or placed anywhere on the screen.

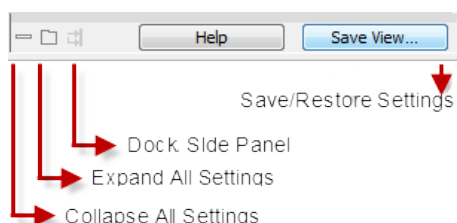


Figure 2.12: Functionalities found at the bottom of the Side Panel.

- Expand 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. Changes made to the Side Panel, including the organization of palettes, will not be saved when you save the view. Learn how to save Side Panel settings in section 4.6.

2.2 Zoom and selection in View Area

All views except tabular and text views support zooming. Figure 2.13 shows the zoom tools, 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.

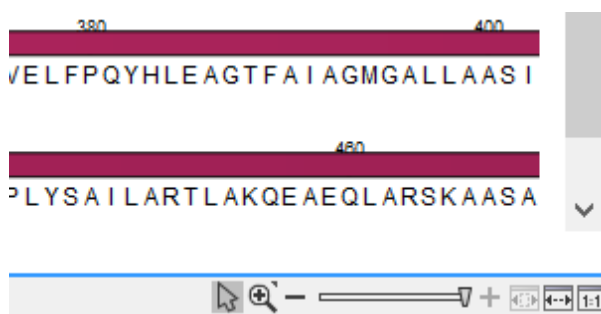


Figure 2.13: The zoom tools are located at the bottom right corner of the view.

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 [12.2](#).

2.2.1 Zoom in

There are six ways of **zooming in**:

- or **Click Zoom in mode** (🔍) **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 mode** (🔍) **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 the data at base level, click the **Zoom to base level** (1:1) icon.

2.2.2 Zoom out

It is possible to zoom out in different ways:

- or **Click Zoom out mode** (🔍) **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.

2.2.3 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 2.2.1. If you press and hold this button, two other modes become available as shown in figure 2.14:

- **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 2.14: Additional mouse modes can be found in the zoom tools when right-clicking on the magnifying glass.

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.

2.3 Toolbox and Status Bar

The **Toolbox** is placed in the left side of the user interface of *CLC Main Workbench* below the Navigation Area. It can be seen as a **Processes tab**, a **Toolbox tab** and a **Favorites tab**.

The Toolbox can be hidden, so that the Navigation Area is enlarged:

Click the **Hide Toolbox** (☞) button or

View | Show/Hide Toolbox

This path gives you the choice to hide the Toolbox, or to selectively hide any of the tabs associated to the Toolbox.

2.3.1 Processes

Click on the **Processes** tab to select it. In this tab, running Workbench processes and any processes previously run in the Workbench session are shown. Server processes are also shown, as described further below.

Running Workbench processes paused and resumed, or stopped. These actions are taken by clicking the small icon (▾) next to the process (see figure 2.15).

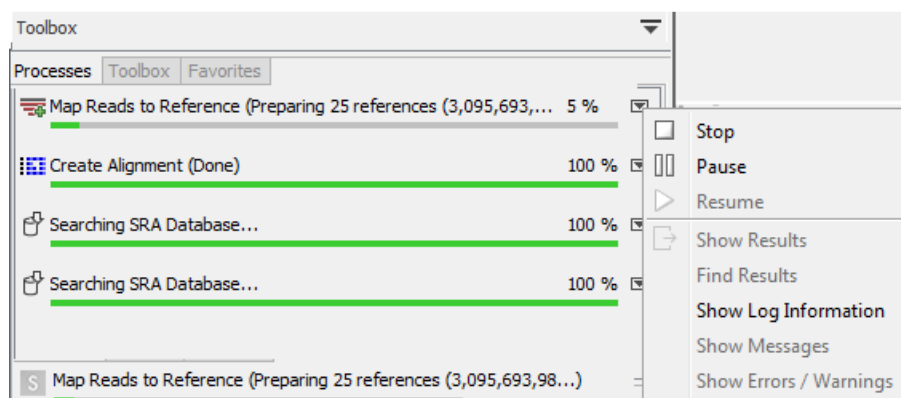


Figure 2.15: A database search and an alignment calculation are running. Clicking the small icon next to the process allow you to stop, pause and resume processes.

Stopped and paused processes are not deleted from the Processes tab during a Workbench session, but they can be removed by right clicking in the Processes tab and selecting the option "Remove Finished Processes" or by going to the option in the main menu system:

View | Remove Finished Processes (X)


Other menu options that appear when the small icon (▾) next to a process in the Processes tab is clicked are:


- **Show results.** If you have chosen to save the results (see section 8.2), 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 8.2), you will be able to highlight the results in the Navigation Area.
- **Show Log Information.** Clicking on this option opens a log of the progress of the process. This is the same log that opens if the option **Open Log** option is selected when launching a task.
- **Show Messages.** Analyses may produce messages when processing your data. Such messages appear briefly in black dialogs in the lower left corner of the Workbench. Clicking on this option shows these messages again.


If you close the program while Workbench processes are still running, a dialog will ask if you are sure that you want to close the program. Closing the program will stop these processes and they

cannot be directly restarted when you re-open the Workbench. You would need to launch any tasks interrupted this way again. Running server processes are not stopped, as described below.

Processes submitted to a CLC Server



Processes you have submitted to a *CLC Server* are listed in the Processes tab when the Workbench is logged into the server. Such processes have a server icon () to their left, rather than icons specific to the analysis being run. Processes that are queued or running on a *CLC Server* will reappear in the Workbench processes tab if you restart the Workbench (and log into the server). *CLC Server* processes already finished when you close the Workbench will not be shown again in the processes tab when you restart your Workbench.

Like running Workbench processes, processes running on a *CLC Server* can be stopped, by selecting clicking the small icon () next to the process and selecting the option "Stop". However, unlike jobs running on a Workbench, they cannot be paused or resumed.

Of note when running jobs on a *CLC Server*: If you choose the option "On my local disk or a place I have access to" when launching an import task, then the Workbench must maintain its connection to the *CLC Server* during the first part of the import process, data upload. If you try to close the Workbench during this phase, you will see a warning dialog. You can see what stage tasks are at in the **Processes** tab. Data upload from the Workbench to the server runs as a local, Workbench process. When the upload stage is complete, a new process for the import is started. This import process will have a server icon () to the left of it. At this point, you can disconnect or close your Workbench without affecting the import.

2.3.2 Toolbox

The tools in the toolbox can be accessed by double-clicking, right clicking and choosing "Run", or by dragging elements from the Navigation Area to an item in the Toolbox.

In addition, a **Launch** button () enables quick launch of tools in *CLC Main Workbench*. You can also press Ctrl + Shift + T ( + Shift + T on Mac) to show the quick launch dialog (see figure 2.16).

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. We also match newer tools to older names when the names were updated from one version of the Workbench to the other.

In the example shown in figure 2.17, typing `create` shows a list of tools involving the word "create", and the arrow keys or mouse can be used for selecting and starting a tool.

2.3.3 Favorites

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 2.18.

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.

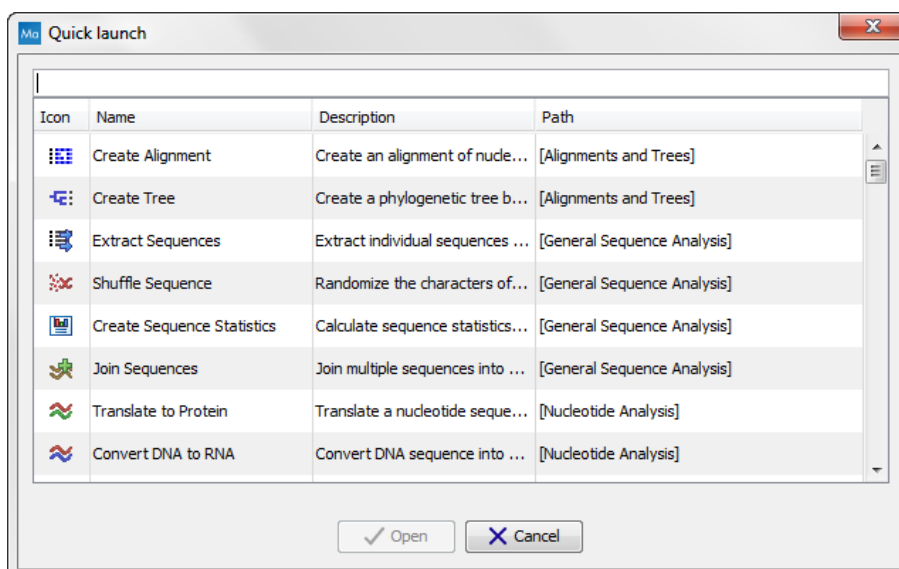


Figure 2.16: Quick access to all tools with Quick Launch.

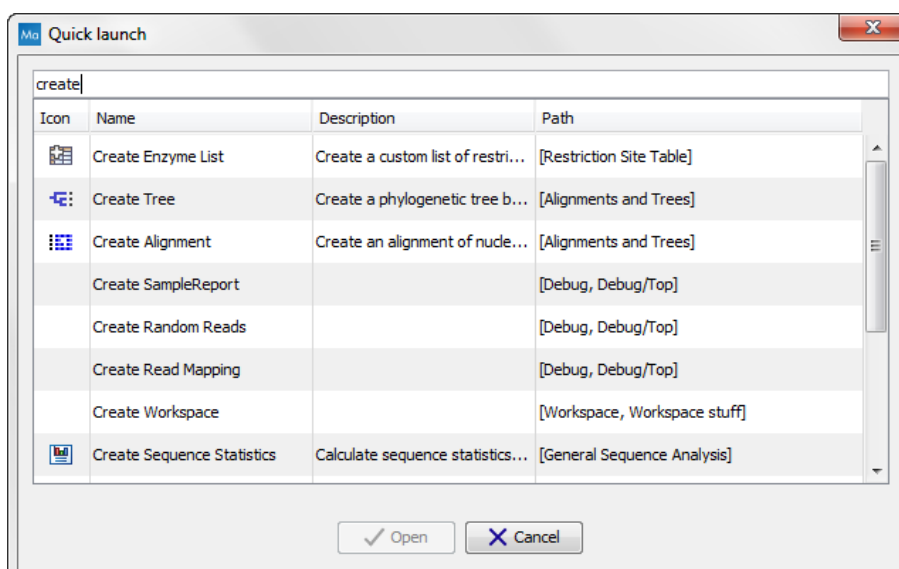


Figure 2.17: Typing in the search field at the top will filter the list of tools to launch.

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.

2.3.4 Status Bar

As can be seen from figure 2.1, 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 various information depending on the context: it can be the size of a region selected on a sequence, the variant at the position where the cursor stands, or how many rows are selected in a table.

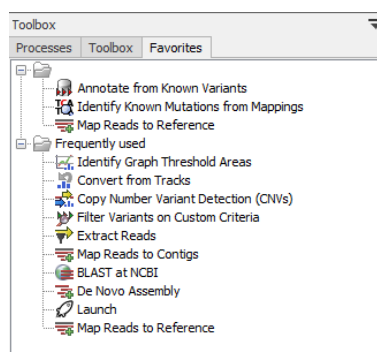


Figure 2.18: Favorites toolbox.

2.4 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.

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

Initially, the folders of the **Navigation Area** are collapsed and the View Area is empty and ready to work with.

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**

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.

2.5 List of shortcuts

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

Action	Windows/Linux	macOS
Adjust selection	Shift + arrow keys	Shift + arrow keys
Adjust workflow layout	Shift + Alt + L	⌘ + Shift + Alt + L
Back to Navigation Area	Alt + Home or Alt + fn + left arrow	⌘ + Home or ⌘ + fn + left arrow
BLAST	Ctrl + Shift + L	⌘ + Shift + L
BLAST at NCBI	Ctrl + Shift + B	⌘ + Shift + B
Close	Ctrl + W	⌘ + W
Close all views	Ctrl + Shift + W	⌘ + Shift + W
Copy	Ctrl + C	⌘ + C
Create alignment	Ctrl + Shift + 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
Launch tools	Ctrl + Shift + T	⌘ + Shift + T
Maximize/restore View size	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
Save	Ctrl + S	⌘ + S
Save As	Ctrl + Shift + S	⌘ + Shift + 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 (one)	⌘ + 1 (one)
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
Switch tabs in View Area	Ctrl + PageUp/PageDown or Ctrl + fn + arrow up/down	Ctrl + PageUp/PageDown or Ctrl + fn + arrow up/down
Switch views	Ctrl + Shift + PageUp/arrow up Ctrl + Shift + PageDown/arrow down	Ctrl + Shift + PageUp/arrow up Ctrl + Shift + PageDown/arrow down
Translate to Protein	Ctrl + Shift + P	⌘ + Shift + P
Undo	Ctrl + Z	⌘ + Z
Update folder	F5	F5
User Preferences	Ctrl + K	⌘ + ,

Scroll and Zoom shortcuts

Action	Windows/Linux	macOS
Vertical scroll in reads 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	Ctrl + Scroll wheel	⌘ + Scroll wheel
Zoom	Ctrl + Scroll wheel	⌘ + 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)
Zoom to base level	Ctrl + 0	⌘ + 0
Zoom to fit screen	Ctrl + 6	⌘ + 6
Zoom to selection	Ctrl + 5	⌘ + 5
Reverse zoom mode	press and hold Shift	press and hold Shift

Workflows related shortcuts

Action	Windows/Linux	macOS
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

Combinations of keys and mouse movements

Action	Windows/Linux	macOS	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 not grouped together	Ctrl	⌘	Click elements
Select multiple elements grouped together	Shift	Shift	Click elements
Select Editor and highlight the corresponding element in the Navigation Area	Alt or Ctrl	⌘	Click tab

"Elements" in this context refers to elements and folders in the **Navigation Area** selections on sequences, and rows in tables.

Chapter 3

Data management and search

Contents

3.1	Navigation Area	60
3.1.1	Data structure	60
3.1.2	Create new folders	63
3.1.3	Sorting folders	63
3.1.4	Multiselecting elements	64
3.1.5	Moving and copying elements	64
3.1.6	Change element names	65
3.1.7	Delete, restore and remove elements	66
3.1.8	Show folder elements in a table	67
3.2	Metadata	69
3.2.1	Importing Metadata	70
3.2.2	Advanced Metadata Import	71
3.2.3	Associating data elements with metadata	76
3.2.4	Working with data and metadata	81
3.3	Working with tables	84
3.3.1	Filtering tables	85
3.4	Customized attributes on data locations	87
3.4.1	Filling in values	90
3.4.2	What happens when a clc object is copied to another data location?	91
3.4.3	Searching	91
3.5	Local search	92
3.5.1	Quick search	93
3.5.2	Advanced search	95

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.

3.1 Navigation Area

The **Navigation Area** (see figure 3.1) is used for organizing and navigating data. Its behavior is similar to the way files and folders are usually displayed on your computer.

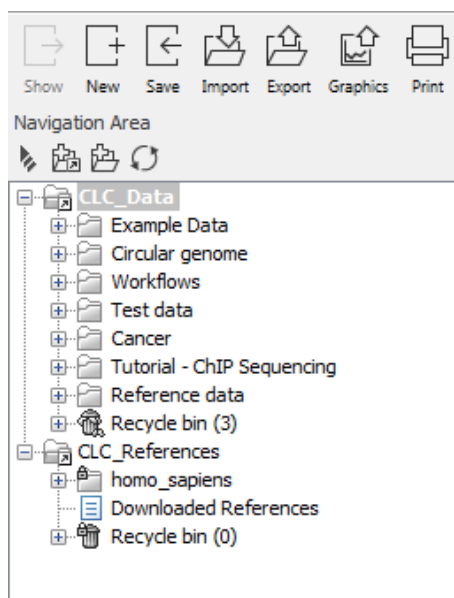


Figure 3.1: *The Navigation Area.*

Just above the area with the listing of data are 4 icons. From left to right, these are:

- **Collapse all** (). This closes all the open folders in the Navigation Area.
- **Add File Location** (). This is explained in section 3.1.1.
- The Create Folder icon (), which is used to create new folders within a configured File Location.
- The Update All icon (), which refreshes the view of the Navigation Area.

To provide more space for viewing data, you can hide **Navigation Area** and the **Toolbox** by clicking on the hide icon () in the top right hand side of the Navigation Area.

3.1.1 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 Workbench data location represents a folder on the computer: The data shown under a Workbench location in the **Navigation Area** is stored on the computer, in the folder the location points to.

This is explained visually in figure 3.2. The full path to the system folder can be seen by mousing over the data location folder icon as shown in figure 3.3.

Data held on a CLC Server

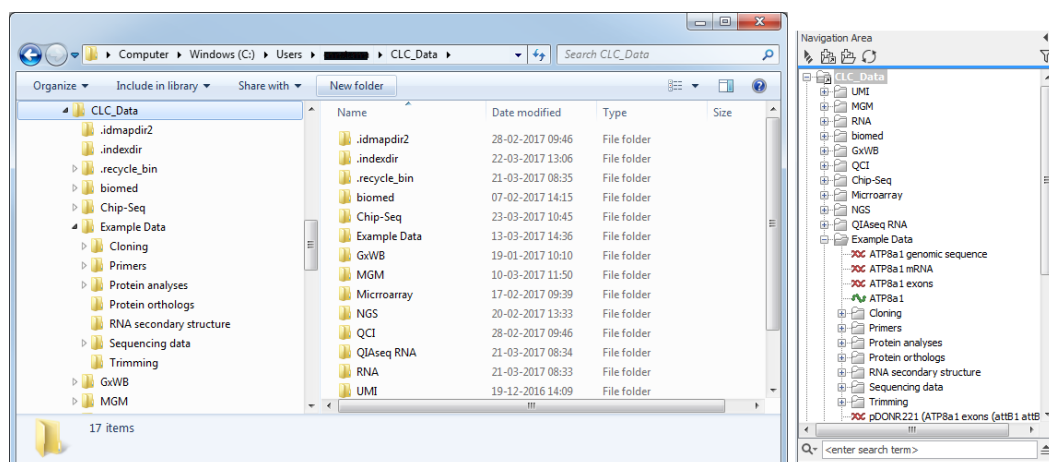


Figure 3.2: In this example the location called "CLC_Data" points to the folder at `C:\Users\\CLC_Data`.

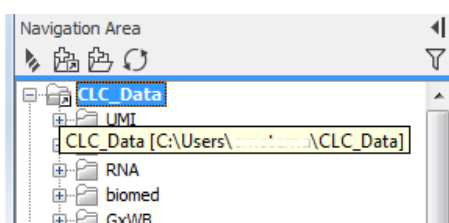


Figure 3.3: Mousing over the location called 'CLC_Data' shows the full path to the system folder, which in this case is `C:\Users\\CLC_Data`.

If you have logged into a CLC Server from your Workbench, then data stored on the CLC Server will also be listed in the Workbench Navigation Area, as illustrated in figure 3.4.

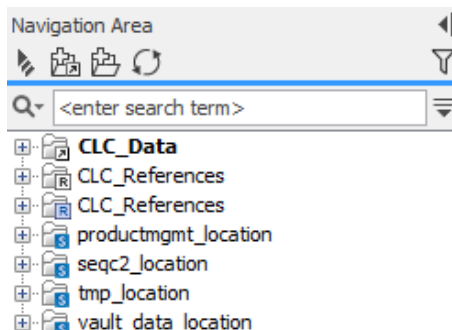


Figure 3.4: Data areas on the CLC Server are highlighted with blue square icons in the Navigation Area.

Adding locations

Per default, there is one location in the **Navigation Area** called CLC_Data. It points to the following folder:

- On Windows: `C:\Users\\CLC_Data`
- On Mac: `~/CLC_Data`
- On Linux: `/homefolder/CLC_Data`

You can easily add more locations to the **Navigation Area**:

File | New | Location (📁)

This will bring up a dialog where you can navigate to the folder you wish to use as your new location (see figure 3.5).

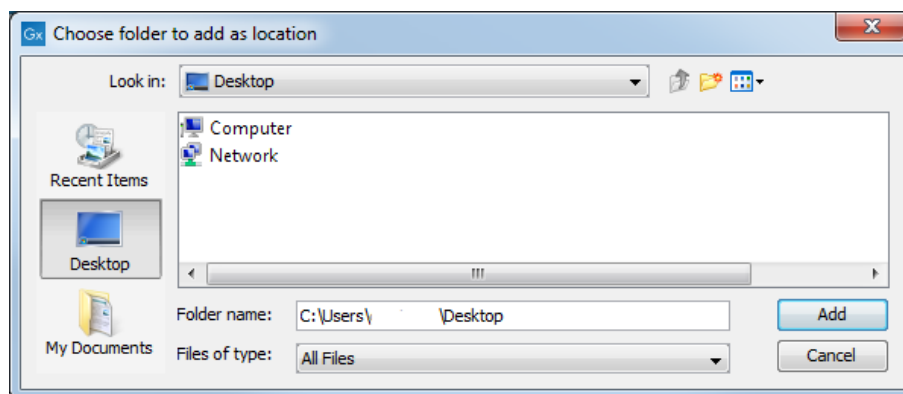


Figure 3.5: Navigating to a folder to use as a new location.

When you click **Open**, the new location is added to the **Navigation Area** as shown in figure 3.6.

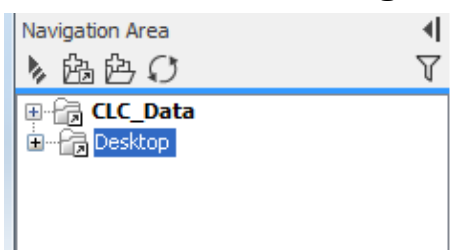


Figure 3.6: The new location has been added.

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 a second. This will show the path to the location.

You can configure any folder on a network drive or a removable drive as a File Location. Such a location will appear inactive if the relevant drive is not available when you start up the Workbench. Once the drive is available, click on the Update All symbol (🔄) at the top of the Navigation area. This refreshes the view of the Navigation Area, and all available Locations will then be shown as active. There can be sometimes be a short delay before the interface update completes.

Sharing data is possible when a network drive is available to multiple Workbenches. In this case, you can add the same folder as a Data Location on each Workbench. However, it is important to note that data sharing is not actively supported: we do not support concurrent alteration of data and while the software will often detect this situation and handle it appropriately, by for example only allowing read access to all but the one party editing the file, we do not guarantee this. In addition, any functionality that involves using the data search indices, (e.g. search functionality, associating metadata with data), will not work properly for shared data locations. Re-indexing a Data Location can help in the short term, but as soon as a new file is created by another piece of software, the index will be out of date. If you decide to share data via Workbenches this way, it is vital that any Workbench that adds a Data Location already used by other Workbenches uses as a Data Location **the exact same folder from the network drive file system hierarchy as the other Workbenches have used**. Indicating a folder higher up or lower down in the hierarchy will

cause problems with the indexing of the files, meaning that newly created objects by Workbench A will not be found by Workbench B and vice versa.

Opening data

The elements in the **Navigation Area** are opened by:

Double-clicking on the element

or **Clicking once on the element | Show () in the Toolbar**

or **Clicking once on the element | Right-click on the element | Show ()**

or **Clicking once on the element | Right-click on the element | Show (the one without an icon) | Select the desired way to view the element from the menu that appears when mousing over "Show"**

This will open a view in the **View Area**, which is described in section [2.1](#).

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 [6](#)). Furthermore, an element can be added by dragging it into the **Navigation Area**. This could be views that are open, elements on lists, e.g. search hits or sequence lists, and files located on your computer.

Finally, you can add data by adding a new location (see section [3.1.1](#)).

If a file or another element is dropped on a folder, it is placed at the bottom of the folder. If it is dropped on another element, it will be placed just below that element.

If the element already exists in the **Navigation Area** a copy will be created with the name extension "-1", "-2" etc. if more than one copy exist.

3.1.2 Create new folders

In order to organize your files, they can be placed in folders. Creating a new folder can be done in two ways:

right-click an element in the Navigation Area | New | Folder ()

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

If a folder is selected in the **Navigation Area** when adding a new folder, the new folder is added at the bottom of this folder. If an element is selected, the new folder is added right above that element.

You can move the folder manually by selecting it and dragging it to the desired destination.

3.1.3 Sorting folders

You can sort the elements in a folder alphabetically:

right-click the folder | Sort Folder

On Windows, subfolders will be placed at the top of the folder, and the rest of the elements will be listed below in alphabetical order. On Mac, both subfolders and other elements are listed together in alphabetical order.

3.1.4 Multiselecting elements

Multiselecting elements means that you select more than one element at the same time. This can be done in the following ways:

- Holding down the <Ctrl> key (⌘ on Mac) while clicking on multiple elements selects the elements that have been clicked.
- Selecting one element, and selecting another element while holding down the <Shift> key selects all the elements listed between the two locations (the two end locations included).
- Selecting one element, and moving the cursor with the arrow-keys while holding down the <Shift> key, enables you to increase the number of elements selected.

3.1.5 Moving and copying elements

Elements can be moved and copied in several ways:

- Using **Copy** (□), **Cut** (✂) and **Paste** (□) from the **Edit** menu.
- Using Ctrl + C (⌘ + C on Mac), Ctrl + X (⌘ + X on Mac) and Ctrl + V (⌘ + V on Mac).
- Using **Copy** (□), **Cut** (✂) and **Paste** (□) in the **Toolbar**.
- Using drag and drop to move elements.
- Using drag and drop while pressing Ctrl / Command to copy elements.

In the following, all of these possibilities for moving and copying elements are described in further detail.

Copy, cut and paste functions

Copies of elements and folders can be made with the copy/paste function which can be applied in a number of ways:

select the files to copy | right-click one of the selected files | Copy (□) | right-click the location to insert files into | Paste (□)

or **select the files to copy | Ctrl + C (⌘ + C on Mac) | select where to insert files | Ctrl + P (⌘ + P on Mac)**

or **select the files to copy | Edit in the Menu Bar | Copy (□) | select where to insert files | Edit in the Menu Bar | Paste (□)**

If there is already an element of that name, the pasted element will be renamed by appending a number at the end of the name.

Elements can also be moved instead of copied. This is done with the cut/paste function:

select the files to cut | right-click one of the selected files | Cut (✂) | right-click the location to insert files into | Paste (📄)

or **select the files to cut | Ctrl + X (⌘ + X on Mac) | select where to insert files | Ctrl + V (⌘ + V on Mac)**

When you have cut the element, it is "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 2.1.6).

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

3.1.6 Change element names

This section describes two ways of changing the names of sequences in the **Navigation Area**. In the first part, the sequences themselves are not changed - it's their representation that changes. The second part describes how to change the name of the element.

Change how sequences are displayed

Sequence elements can be displayed in the **Navigation Area** with different types of information:

- Name (this is the default information to be shown).
- Accession (sequences downloaded from databases like GenBank have an accession number).
- Latin name.
- Latin name (accession).
- Common name.
- Common name (accession).

Whether sequences can be displayed with this information depends on their origin. Sequences that you have created yourself or imported might not include this information, and you will only be able to see them represented by their name. However, sequences downloaded from databases like GenBank will include this information. To change how sequences are displayed:

right-click any element or folder in the Navigation Area | Sequence Representation | select format

This will only affect sequence elements, and the display of other types of elements, e.g. alignments, trees and external files, will be not be changed. If a sequence does not have this information, there will be no text next to the sequence icon.

Rename element

Renaming a folder or an element in the **Navigation Area** can be done in 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 [11.3.3](#).


3.1.7 Delete, restore and remove elements

When one deletes data held in a Workbench data location, it is moved to the recycle bin within 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 [11.3.4](#).

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.

Deleting data held on a CLC Server



You can delete data that you have "write" permission for from *CLC Server* data areas when logged into a server from your Workbench. The method of deleting data is the same as described above when deleting data held in Workbench data locations. The deleted data is placed in a **Recycle bin** () on the *CLC Server*. The data in the server-based recycle bin can only be accessed by you and the server administrator. Note that the server administrator may have configured the recycle bin to be automatically emptied at regular intervals.

3.1.8 Show folder elements in a table

A location or a folder might contain large amounts of elements. It is possible to view their elements in the **View Area**:

select a folder or location | Show () **in the Toolbar**

or

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

An example is shown in figure [3.7](#).

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 (\mathbb{C} 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.

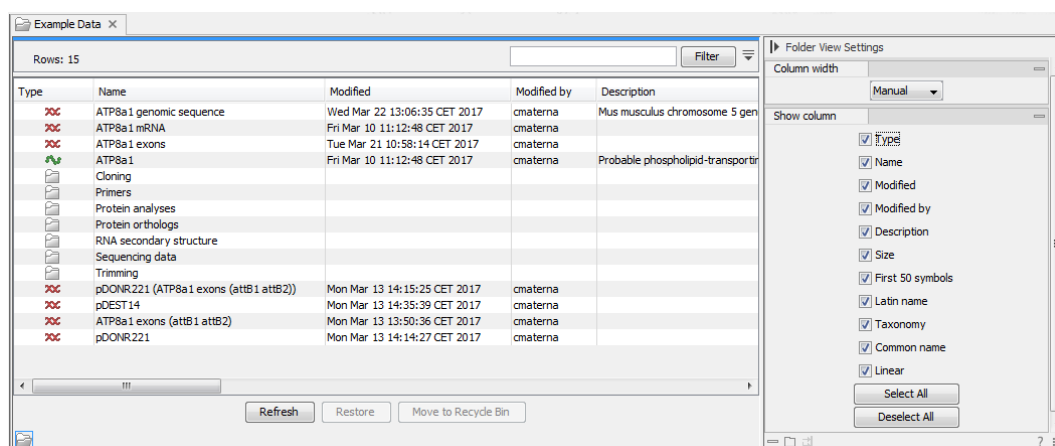


Figure 3.7: Viewing the elements in a folder.

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 3.8 you can see an example where the name of two sequences 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.

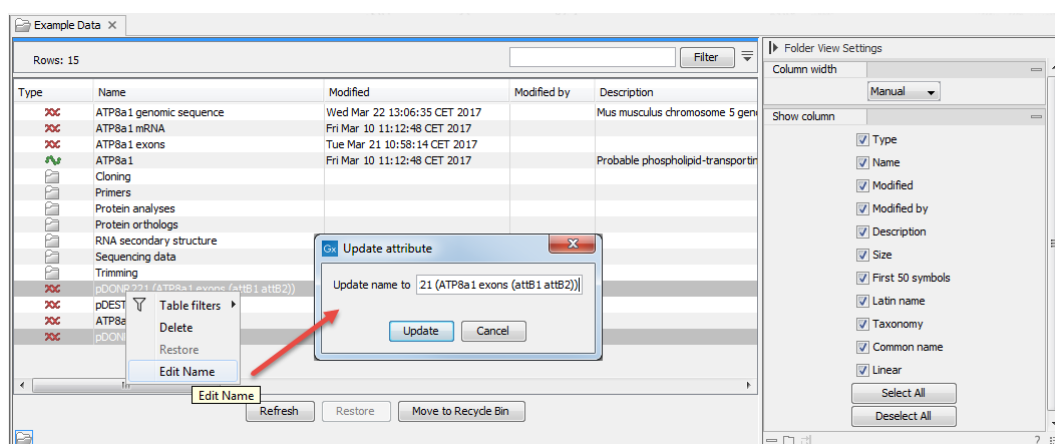


Figure 3.8: Changing the common name of two sequences.

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

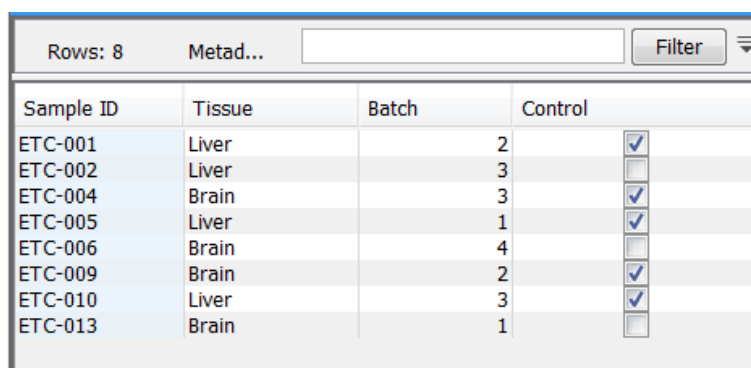
Drag and drop folder elements

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.

3.2 Metadata

Metadata refers to information about data. In the context of the CLC Workbenches, this will usually mean information about samples. For example a set of reads could come from a particular specimen at a particular time point with particular characteristics. The specimen, time and characteristics would be metadata for that set of reads. The data can then be associated with its metadata in the Workbench. This can be useful for keeping track of related datasets and metadata can be used by some types of analyses in some CLC Workbenches.

Metadata can be created directly in the Workbench, but typically it will be imported from an external file (excel or text based). See section 3.2.1. It is then stored as a metadata table in the Workbench. An example of a metadata table as it might appear in the Workbench is shown in figure 3.9.



Sample ID	Tissue	Batch	Control
ETC-001	Liver	2	<input checked="" type="checkbox"/>
ETC-002	Liver	3	<input type="checkbox"/>
ETC-004	Brain	3	<input checked="" type="checkbox"/>
ETC-005	Liver	1	<input checked="" type="checkbox"/>
ETC-006	Brain	4	<input type="checkbox"/>
ETC-009	Brain	2	<input checked="" type="checkbox"/>
ETC-010	Liver	3	<input checked="" type="checkbox"/>
ETC-013	Brain	1	<input type="checkbox"/>

Figure 3.9: A simple Metadata Table.

Each column represents a property of a sample (e.g. identifier, height, age, treatment, etc.) and each row contain information relevant to a sample.

Within the CLC Workbench, one of the metadata table columns may be designated as the key column. The entries in a key column must be unique. Any column can be chosen to be the key column, but commonly it will be the first column and it would contain an identifier of some sort (e.g. a name).

There are no restrictions on the type of information that can be held in a metadata table. However, it is generally recommended that any given metadata table contains information about a related collection of entities. For example, a set of samples from the same experiment, or a set of families from the same study. Any particular data element can only be associated with *at most one* row in a given metadata table. However, that same data element can be associated with metadata in other metadata tables.

During or after metadata import, data can be associated with that metadata. Once a data element is associated with metadata, the outputs of analyses involving that data usually inherit the metadata association automatically. Inheritance like this is carried out when the metadata association for the outputs can be unambiguously identified. So, for example, if an output is derived from two inputs with different metadata associations, then neither association will be inherited by the output data elements.

Importing metadata can be done using a basic or advanced tool, and viewing and working with metadata, including data association, is done using the Metadata Table editor.

3.2.1 Importing Metadata

There are two tools that can be used to import metadata, one basic and one more advanced. A list of the benefits and limitations of each is included at the start the sections describing them.

The basic import tool is fast and easy, but less flexible than the advanced metadata import using the Metadata Table Editor. General features of this importer are:

- Excel (.xlsx/.xls) format files are imported.
- The first column in the Excel file must have unique entries. That column is designated the key column.
- Optionally, data elements can have associations to the metadata made.
- Metadata association using this tool matches data element names with the entries in the first column of the metadata being imported. Name matching can be based on exact or partial matches.
- Data elements that will be associated to metadata being imported are listed in a preview window.
- All columns are imported as text columns.

If desired, a metadata table can be edited later from within the Metadata Table editor as described in section 3.2.3. There, you can change the column data types (e.g. to types of numbers, dates, true/false) and you can designate a new key column.

To run the basic importer, go to:

File | Import (📄) | Import Metadata (📊)

In the box labeled **Spreadsheet with sample information**, select the Excel file (.xlsx/.xls) to be imported.

The rows in the spreadsheet are displayed in the Metadata preview window, as shown in figure 3.10. Click **Next**.

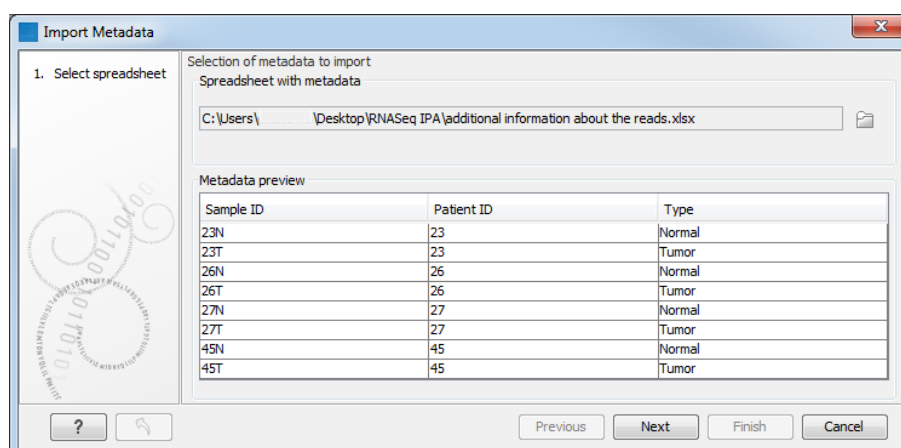


Figure 3.10: After an Excel file is selected, its rows are visible in the Metadata preview table.

The second wizard step, called "Associate with data", is optional. To proceed without associating data to metadata, click on the the button labeled **Next**.

Associating data with the metadata being imported is illustrated in figure 3.11. To do this:

- In the field labeled **Location of data**, click on the folder icon to the right and select the data elements of interest.
- In the Matching scheme section, select whether data element names must match exactly the entries in the first column of the metadata to have an association created (Exact), or whether partial matches are allowed (Partial). The two matching schemes are described in detail in section 3.2.3.

The Data association preview area shows data elements that will have associations created, along with information from the metadata row they are being linked with. This gives the opportunity to check that the matching is leading to the expected links between data and metadata.

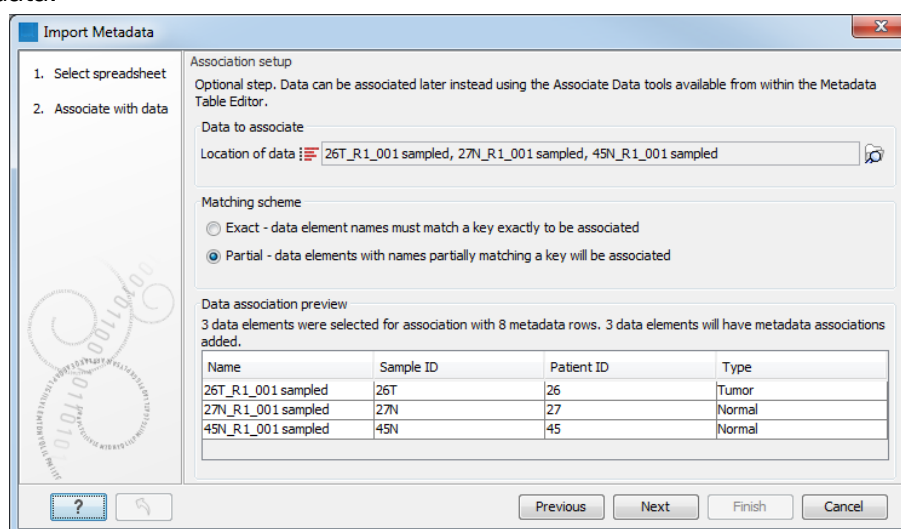


Figure 3.11: Three data elements were selected for association with 8 metadata rows. All three will have an associated added. Here, the partial matching scheme has been selected.

- Click on the button labeled **Next**.
- Select where you wish the metadata table to be saved.
- Click on the button labeled **Finish**.

The associated information can be viewed for a given data element in the Show Element Info view, as show in figure 3.12.

3.2.2 Advanced Metadata Import

If the information about the data is in an excel file and the entries in the first column are unique, then the Import Metadata tool described in section 3.2.1 can be used to define the table and import the metadata in a couple of steps.

In other cases, the **Metadata Table Editor** can be used to import metadata from an external file, or to create and populate a metadata table directly. It involves more steps than the basic import tool, but is more flexible and has some basic error checking associated with data types. General features of this importer are:

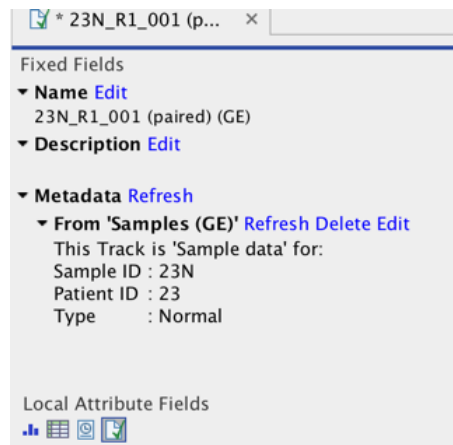


Figure 3.12: Metadata associations can be seen, edited, refreshed or deleted via the Show Element Info view.

- Can import from Excel (.xlsx/.xls) or text files with a common delimiter can be used.
- The structure of the metadata table (the columns, their type, and the key column) must be set up before the metadata (contents) are imported.
- It is generally recommended that one column be designated as the key column. Entries in that column must have unique entries.
- The default data type for columns on creation is text, but this can be altered before import commences. When importing the metadata, an error will result if entries are found that do not match the expected data type.
- Association with metadata is done by matching data element names with the entries in the first column of the spreadsheet. Name matching can be based on exact or partial matches.
- Association of data with metadata is done as a separate step from import, providing flexibility. For example, if information in more than one column together uniquely identifies a sample, but the information within a single given column does not uniquely do so.
- Association of data with metadata can be done row by row if key column entries and the names of the relevant data elements are not related.

To start the Metadata Table Editor, go to:

File | New | Metadata Table (📄)

This opens a new metadata table with no columns and no rows. Importing metadata using the Metadata Table Editor requires that the **table structure** is defined first.

Defining the table structure

Click on the button labeled **Setup Table** at the bottom of the view (figure 3.13).

To create a metadata table from scratch, use the "Add column right" or "Add column left" buttons (📄) to define the table structure with the amount of columns you will need, and edit the fields of each column as needed.

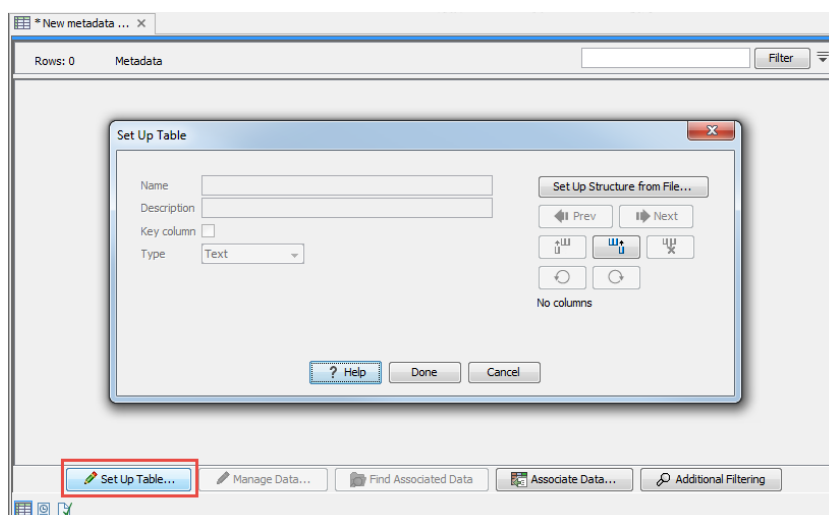


Figure 3.13: Dialog used to add columns to an empty Metadata Table.

To import the table from a file, click on **Setup Structure from File**. In the dialog that appears (figure 3.14), you need to provide the following information:

- **Filename** The EXCEL or delimited TEXT file to import. Column names should be in the first row of this file.
- **Encoding** For text files only: the encoding used to create the file. The default is UTF-8.
- **Separator** For text files only: The character used to separate the columns. The default is semicolon (;).

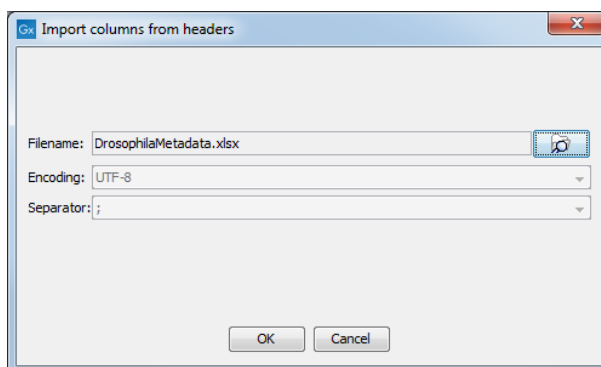


Figure 3.14: Creating a metadata table structure based on an external file.

For each column in the external file, a column will be created in the new metadata table. By default the type of these imported columns is "Text". You will see a reminder to set the column type for each column and to designate one of the columns as the key column.

Edit the following information for each column:

- **Name**. A mandatory header name or title for the column.
- **Description**. An optional description of the information that will be held in the column. The description will appear as a tool tip, visible when you hover the mouse cursor over the column name in the metadata table.

- **Key column.** Put a check in the box in the one column that will be the "key" column. All rows in this column must be populated and all entries in this column must be unique.
- **Type.** The type of value allowed. The available types are:
 - **Text** Simple text.
 - **Whole number** Integer values, like 42 or -7.
 - **Decimal number** Decimal values, like 3.14 or 1.72e13.
 - **Yes / No** Yes/No or True/False values are accepted. Capitalization is not necessary.
 - **Date** Local dates such as 2015-04-23 for April 23rd, 2015.
 - **Date and time** Local date and time such as 2015-04-23 13:37 for 1:37pm on April 23rd, 2015. Note the use of 24-hour clock and that no time zone information is present.

Navigate between the columns using the (◀) Prev and (▶) Next buttons, or by using left/right arrow keys with Alt key held down.

Modifications made to a particular column take effect as you navigate to another column, or if you close the dialog using **Done**.

The (↶) and (↷) buttons are used undo and redo changes respectively. When the columns have been configured, click on the button labeled **Done**.

Columns may be deleted using the (✖) button. After metadata has been imported, additional columns can be added to the table structure. This can be done by importing the altered structure from an external file, where any columns not already in the metadata table will be added. Alternatively, individual columns can be added using the (↑) and (↓) buttons, which insert new columns before and after the current column respectively.

Populating the table

Click on **Manage Data** button at the bottom of the view (figure 3.15).

The metadata table can then be populated by editing each column manually. Row information is added manually by clicking on the (⊞) button and typing in the information for each column.

It is also possible to import information from an external file. In that case, the column names in the metadata table in the workbench will be matched with those in the external file to determine which values go into which cell. Only cell values in columns with an exact name match will be imported. If the file used contains columns not in the metadata table, the values in those columns will be ignored. Conversely, if the metadata table contains columns not present in the file, imported rows will have no values for those columns.

Click on **Import Rows from File** and select the external file of metadata. This brings up the window shown in figure 3.16.

When working with an existing metadata table and adding extra rows, it is generally recommended that a key column be designated first. If a key column is not present, then all rows in the file will be imported. With no key column designated, if any rows from that file were imported into the same metadata table earlier, a duplicate row will be created. With a key column, rows with

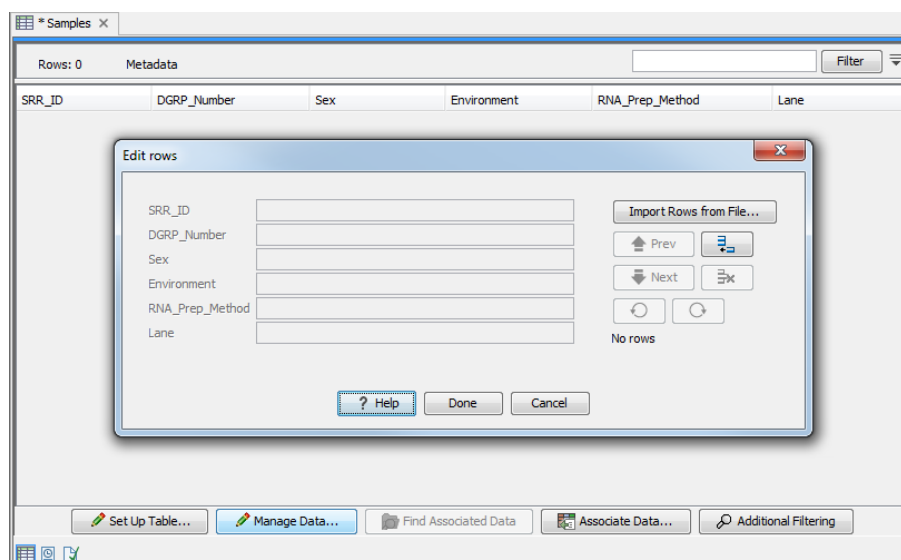


Figure 3.15: Tool for managing the metadata itself. Notice the button labeled *Import Rows from File*.

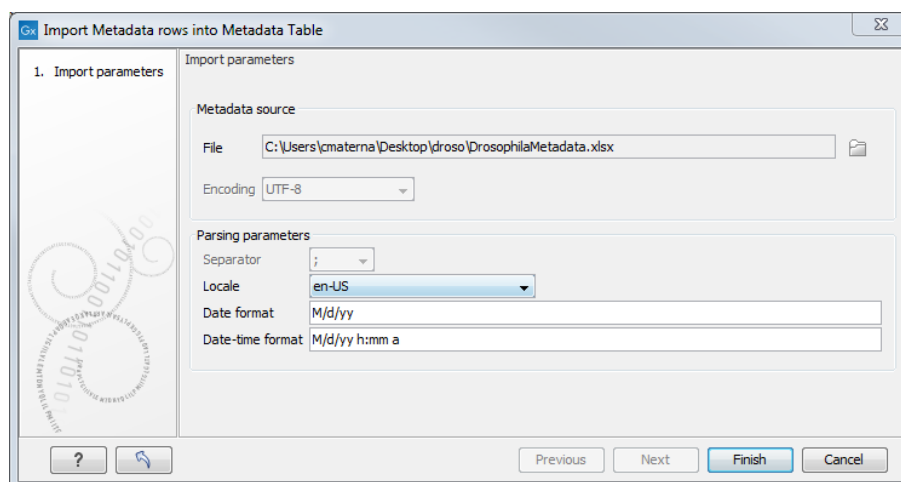


Figure 3.16: Tool to import rows into a *Metadata Table*.

a new, unique entry for that column are added to the table and existing rows with a key entry in the file will be updated, incorporating any changes present in the file. Duplicate rows will not be created.

The options presented in the *Import Metadata Rows into Metadata Table* are:

- **File.** The file containing the metadata to import. This can be Excel (.xlsx/.xls) format or a delimited text file.
- **Encoding.** For text files only: The text encoding of the selected file. Specifying the correct encoding is important to ensure that the file is correctly interpreted.
- **Separator.** For text files only: the character used to separate columns in the file.
- **Locale.** For text files only: the locale used to format numbers and dates within the file.
- **Date format.** For text files only: the date format used in the imported file.

- **Date-time format.** For text files only: the date-time format used in the imported file. The date and date-time templates uses the Java patterns for date and time formatting. Meaning of some of the symbols:

Symbol	Meaning	Example
y	Year	2004; 04
d	Day	10
M/L	Month	7; 07; Jul; July; J
a	am-pm	PM
h	Hour (0-12 am pm)	12
H	Hour (0-23)	0
m	Minute	30
s	Second	55

Examples of using this:

Format	Meaning	Example
dd-MM-yy	Short date	31-12-15
yyyy-MM-dd HH:mm	Date and Time	2015-11-23 23:35
yyyy-MM-dd'T'HH:mm	ISO 8601 (standard) format	2015-11-23T23:35

With a short year format (YY), 2000 will be added when imported as, or converted to, Date or Date and time format. Thus, when working with dates before the year 2000 or after 2099, please use a four digit format for the year (YYYY).

Click the button labeled **Finish** button when the necessary fields have been filled in.

The progress and status of the row import can be seen in the Processes tab of the Toolbox. Any errors resulting from an import that failed can be reviewed here. The most frequent errors are associated with selecting the wrong separator or encoding, or wrong date/time formats when importing rows from delimited text files.

Once the rows are imported, The metadata table can be saved.

3.2.3 Associating data elements with metadata

Typically, one would use the tools described in this section to associate data elements with metadata just after the data has been imported. Doing this at this early stage means that analysis results generated using these inputs will often inherit the metadata association. This inheritance is done when the relevant association can be determined unambiguously.

Each association between a particular data element and a row in your Metadata Table will have a "role" label that indicates what the role of the data element has. For example, a newly import sequence list could be given a role like "Sample data", or "NGS reads". Each analysis tool provides a particular role label when applying a metadata association to the outputs it generates. For example, a read mapping tool could assign the role "Un-mapped reads" to a sequence list of unmapped reads that it produces. When viewing all the data associated with a given metadata entry, these roles can help distinguish the particular data elements of interest.

The metadata table must be saved before data association options are available to use.

To associate data elements with the rows of a Metadata Table, click the **Associate Data** button at the bottom of the Metadata Table view. When an metadata association is created for, or removed from, a data element, this change to the data element is automatically saved.

If a key column has been identified for the metadata table, two options will be available:

- **Association Data Automatically:** The whole metadata table is used and associations between the selected data elements and the metadata are applied based on matching of the element name with the key column entries in the metadata table.
- **Associate Data with Row:** You select a row of the metadata and a particular data element and an association is then created. Information in the metadata table does not need to match the name of the data elements using this option. This option is also available when right-clicking a row in the table.

Each of these has benefits and restrictions. These are described at the top of each sections describing these options.

Associate Data Automatically

The main characteristics of the **Associate Data Automatically** tool are:

- Suited to associated large metadata tables or associating to many data elements.
- Well suited for use with newly imported data, where no associations already exist.
- Associations are created based on matching the information in the key column of the metadata table with the name of the selected data elements.
- Two matching schemes are available: Exact and Partial (see section [3.2.3](#)).
- A key column must be identified for the metadata table for this option to be available.
- Use with care with data elements that already have associations with the metadata table being worked with. As well as adding any new associations, existing associations will be *updated* to reflect the current information in the metadata table. This means associations will be *deleted* for a selected data element if there are no rows in the metadata table that match the name of that data element. See also the warning at the end of this section about this.

To run the **Associate Data Automatically** tool, click the **Associate Data** button at the bottom of the Metadata Table view, and select **Associate Data Automatically**.

Your metadata table must be saved and a key column designated for the metadata table for this option to be available.

Select the data the tool should consider when setting up metadata associations in the window that appears. An example of this is shown in figure [3.17](#). You can select an item or sets of items in the navigation area on the left and move these into the selected elements list. Alternatively, you can right click on a folder and specify that all elements in the folder should be put in the selected elements list. This is illustrated in figure [3.18](#).

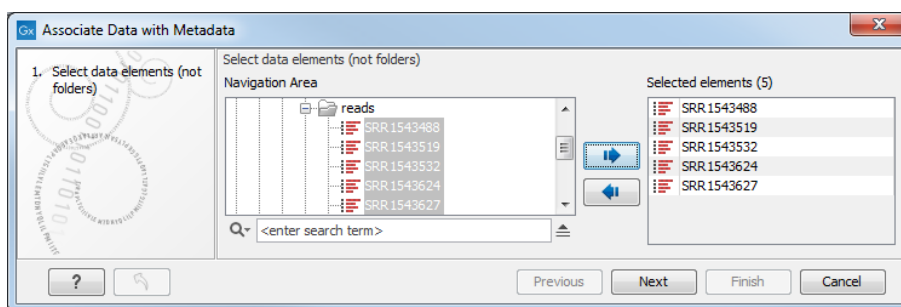


Figure 3.17: Select data for automatic metadata association.

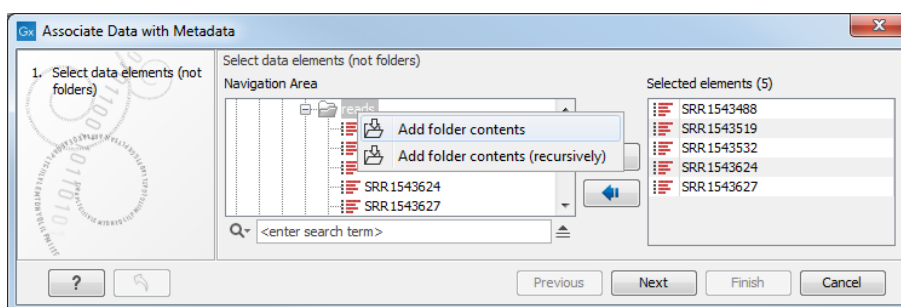


Figure 3.18: Selecting all data elements in a folder.

Specify a role that should be assigned to each data element that is associated to a metadata row (figure 3.19). The role can be anything that describes the data element best.

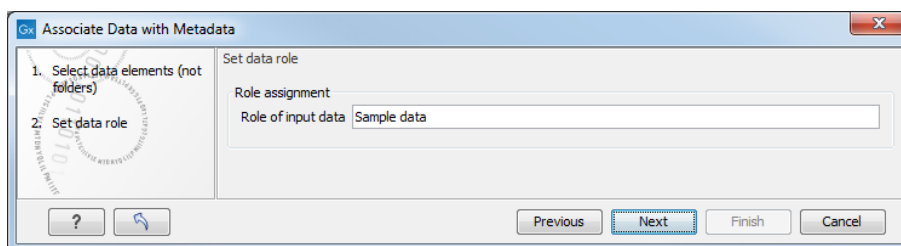


Figure 3.19: Provide a role for the data elements. The default role provided is "Sample data".

Select whether the matching of the data element names to the entries in the key column should be based on exact or partial matching. These options are explained further below.

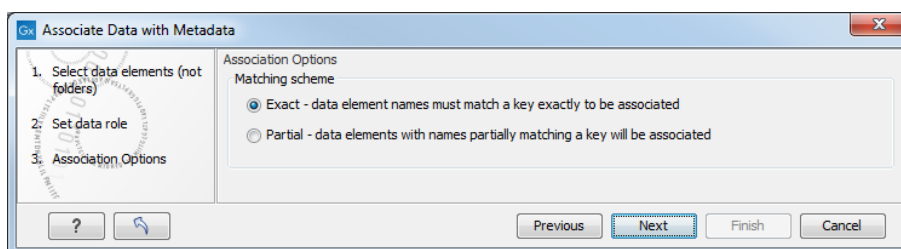


Figure 3.20: Data element names can be matched either exactly or partially to the entries in the key column.

Choose to **Save** the outputs. Data associations and roles will be saved for data elements where the name matches a key column entry according to the selected matching scheme.

Warning: It is safest only to select data elements that have no existing association to the metadata table being worked with, or carefully selecting any data elements with an existing association which you wish to update. All data selected that has an association with the

metadata table being worked with will be *updated* by the automatic association tool. This means that any new or updated information in a metadata row can be added, but it also means that if no rows in the metadata match such a data element anymore, then the data association will be removed. This could happen if, for example, you changed the name of a data element with a metadata association, and did not change the corresponding key entry in the metadata table.

Matching schemes A data element name must match an entry in the key column of a metadata table for an association to be set up between that data element at the corresponding row of the metadata table. Two schemes are available in the **Association Data Automatically** for matching up names with key entries:

- **Exact** - data element names must match a key exactly to be associated. If any aspect of the key entry differs from the name of a selected data element, no association will be created.
- **Partial** - data elements with names partially matching a key will be associated. Here, data element names are broken into parts using common delimiters. The first whole part(s) must match a key entry in the metadata table for an association to be established. This option is explained in more detail below.

Partial matching rules For each data element being considered, the partial matching scheme involves breaking a data element name into components and searching for the best match from the key entries in the metadata table. In general terms, the best match means the longest key that matches entire components of the name.

The following describes the matching process in detail:

- Break the data element name into its component parts based on the presence of delimiters. It is these parts that are used for matching to the key entries of the metadata table.

Delimiters are any non-alphanumeric characters. That is, anything that is not a letter (a-z or A-Z) or number (0-9). So, for example, characters like hyphens (-), plus symbols (+), spaces, brackets, and so on, would be used as delimiters.

If partial matching was chosen with a data element called `Sample234-1 (mapped) (trimmed)` would be split into 4 parts: `Sample234`, `-1`, `(mapped)` and `(trimmed)`.

- Matches are made at the component level. A whole key entry must match perfectly to at least the first complete component of a data element name.

For example, a key entry `Sample234` would be a match to the data element with name `Sample234-1 (mapped) (trimmed)` because the whole key entry matches the whole of the first component of the data element name. Conversely, if the key entry had been `Sample23`, no match would be identified, because the whole key entry does not match to at least the whole of the first component of the data element name.

In cases where a data element could be matched to more than one key, the longest key matched determines the metadata row the data will be associated with.

The table below provides examples to illustrate the partial matching system, on a table that has the keys with sample IDs like in figure 3.21) (i.e. ETC-001, ETC-002, . . . , ETC-013),

Data Element	Key	Reason for association
ETC-001 (Reads)	ETC-001	Key ETC-001 matches the first part of the name
ETC-001 un-m. . . (single)	ETC-001	''
ETC-001 un-m. . . (paired)	ETC-001	''
ETC-002	ETC-002	Key ETC-002 matches the whole name
ETC-003	None	No keys match this data element name
ETC-005	ETC-005	Key ETC-005 matches the whole name
ETC-005-1	ETC-005	Key ETC-005 matches the first part of the name
ETC-006-5	ETC-006	Key ETC-006 matches the first part of the name
ETC-007	None	No keys match this data element name
ETC-007 (mapped)	None	''
ETC-008	None	''
ETC-008 (report)	None	''
ETC-009	ETC-009	Key ETC-009 matches the whole name

Associate Data with Row

The main characteristics of the **Associate Data with Row** tool are:

- Suited for association of a few metadata tables to a few data elements.
- Full control to select which data element should be associated with a particular metadata row.
- No requirement for a key column in the metadata table.
- No requirement for a relationship between the name of the data element and the metadata to associate it with.

To associate data elements with a particular row in the metadata table:

- Select the desired row in the metadata table by clicking on it.
- Right click and select the **Associate Data with Row** option (see figure 3.21), or click on the **Associate Data** button at the bottom of the view and choose the option **Associate Data with Row**.

The screenshot shows a table with 7 rows and 4 columns: Sample ID, Tissue, Batch, and Control. The row for ETC-006 (Brain, Batch 4) is selected. A context menu is open over this row, showing the option 'Associate Data with Row...'. The Control column has checkboxes for each row, with the ETC-006 row's checkbox checked.

Sample ID	Tissue	Batch	Control
ETC-001	Liver		2
ETC-002	Liver		3
ETC-004	Brain		3
ETC-005	Liver		1
ETC-006	Brain		4
ETC-009	Brain		2
ETC-013	Brain		1

Figure 3.21: Manual association of data elements to a metadata row.

- A window will open within which you can select the data elements that should have an association with the metadata row.

If a selected data element already has an association with this particular metadata table, that association will be updated. Associations with any other metadata tables will be left as they are.

- Click **Next**.
- Enter a role for the data elements that have been chosen and click **Next** until you can choose to **Save** the outputs.

Data associations and roles will be saved for the selected data elements.

3.2.4 Working with data and metadata

Finding data elements based on metadata

Using the Metadata Table view you can find data elements associated with rows of the metadata table. From this view, it is possible to launch analyses on selected data.

To find data elements associated with selected metadata rows:

- Select one or more rows of interest in the metadata table.
- Click on the button labeled **Find Associated Data** at the bottom of the view.

A table with a listing of the data elements associated to the selected metadata row(s) will appear (figure 3.22).

The screenshot shows a software interface with two main tables. The top table, titled 'Metadata', has 7 rows and 4 columns: Sample ID, Tissue, Batch, and Control. The bottom table, titled 'Metadata Elements', has 8 rows and 5 columns: Sample ID, Role, Type, Name, and Path. The 'Find Associated Data' button is highlighted in the interface.

Sample ID	Tissue	Batch	Control
ETC-001	Liver		2
ETC-002	Liver		3
ETC-004	Brain		3
ETC-005	Liver		1
ETC-006	Brain		44
ETC-009	Brain		2
ETC-013	Liver		1

Sample ID	Role	Type	Name	Path
ETC-001	Sample data		ETC-001 (Reads)	CLC_Data / Metadata / Example
ETC-001	Sample data		ETC-001 un-mapped reads (single)	CLC_Data / Metadata / Example
ETC-001	Sample data		ETC-001 un-mapped reads (paired)	CLC_Data / Metadata / Example
ETC-009	Un-mapped reads		BC_9_L001_R1 (paired) un-mapped reads [BC_9_L001_R1] (single)	CLC_Data
ETC-009	Un-mapped reads		BC_9_L001_R1 (paired) un-mapped reads [BC_9_L001_R1] (paired)	CLC_Data
ETC-009	Mapping Report		BC_9_L001_R1 (paired) mapping summary report	CLC_Data
ETC-009	Read mapping		BC_9_L001_R1 (paired) mapping	CLC_Data
ETC-009	Sample data		ETC-009	CLC_Data / Metadata / Example

Figure 3.22: Metadata Table with search results

The search results table shows the type, name, and navigation area path for each data element found. It also shows the key entry of the metadata table row with which the element is associated and the role of the data element for this metadata association. In figure 3.22, there are five data elements associated with sample ETC-009. Three are Sequence Lists, two of which have a role that tells us that they are un-mapped reads resulting from the Map Reads to Reference tool.

Clicking the **Refresh** button will re-run the search and refresh the search results table.

Click the button labeled **Close** to close the search table view.

Data elements listed in the search result table can be opened by clicking on the button labeled **Show** at the bottom of the view.

Alternatively, they can be highlighted in the Navigation Area by clicking the **Find in Navigation Area** button.

Analyses can be launched on the selected data elements:

- Directly. Right click on one of the selected elements, choose the menu option **Toolbox**, and navigate to the tool of interest. The data selected in the search results table will be listed as selected elements in the Wizard that appears.
- Via the Navigation area selection. Use the **Find in Navigation Area** button and then launch a tool in the **Toolbox**. The items that were selected in the Navigation area will be pre-selected in the Wizard that is launched.

If no data elements with associations are found and this is unexpected, please re-index the locations your data are stored in. This is described in section 3.5. For data held in a CLC Server location, an administrator will need to run the re-indexing. Information on this can be found in the CLC Server admin manual at http://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Rebuilding_index.html.

Identifying metadata rows without associated data

Using the Metadata Table view you can apply filters using the standard filtering tools shown at the top of the view as well as by using special metadata filtering in the **Additional Filtering** shown at the bottom. Using the special metadata filtering option **Show only Unassociated Rows**, you can filter the rows visible in the Metadata Table view so only the rows to which no data elements are associated are shown. If desired, these rows could then be used to launch one of the tools for associating data, described in section 3.2.3

This filter can be cancelled by clicking on the **Additional Filtering** button again and clicking on the **Show only Unassociated Rows** again. When the filter is active, it has a checkmark beside it. When it is inactive, it does not.

This filter can take a long time if many rows are shown in the table. When working with many rows, it can help if the full table is filtered using the general filters in advance, using the standard filters at the top of the table view. Alternatively you can pre-select some rows and filtering with the Additional filtering option **Filter to Selected Rows**. This filter can be applied multiple times. If the search takes too long, you can cancel it by unselecting the filter from the menu.

This filter can be cancelled by clicking on the **Additional Filtering** button again and clicking on the **Clear Selection Filter** option.

Viewing metadata associations

Metadata associations for a data element are shown using the Element info view, as in figure 3.23.

To show Element Info,

right-click an element in the Navigation Area | Show | Element Info ()

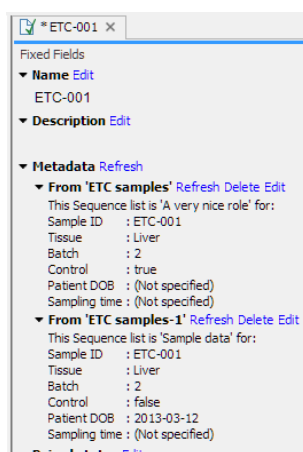


Figure 3.23: *Element Info* view with a metadata association

The Element Info view contains the details of each metadata association for the data element. The following operations are available:

- **Delete** will remove an association.
- **Edit** will allow you to change the role of the metadata association.
- **Refresh** will reload the metadata details from the Metadata Table; this functionality may be used to attempt to re-fetch metadata that was previously unavailable, e.g. due to server connectivity.

Read more about Element Info view in section [11.4](#).

Removing metadata associations

Any or all associations to data elements from rows of a metadata table can be removed by taking the following steps:

1. Open the metadata table containing the rows of interest.
2. Highlight the relevant rows of the metadata table.
3. Click on the button labeled Find Associated Data.
4. In the Metadata Elements table that opens, highlight the rows for the data elements the metadata associations should be removed from.
5. Right click over the highlighted area and choose the option Remove Association(s) (figure [3.24](#)). Alternatively, use the Delete key on the keyboard, or on a Mac, the fn and backspace keys at the same time.

Metadata associations can also be removed from within the Element info view for individual data elements, as described in section [3.2.4](#).

When an metadata association is removed from a data element, this update to the data element is automatically saved.

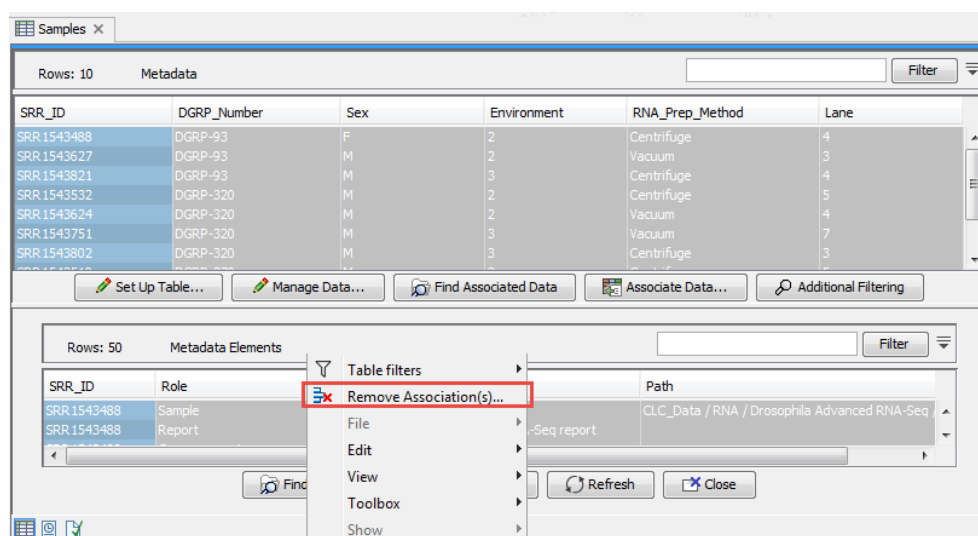


Figure 3.24: Removing metadata associations to two data elements via the Metadata Elements table.

Exporting metadata

The standard Workbench export functionality can be used to export metadata tables to various formats. The system's default locale will be used for the export, which will affect the formatting of numbers and dates in the exported file.

See section 6.2 for more information.

3.3 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 3.25 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.

Sequence	Start	End	Length	Found at strand	Start codon
ATP8a1 genomic sequence	18430	18747	318	positive	ATG
ATP8a1 genomic sequence	19414	19719	306	positive	ATG
ATP8a1 genomic sequence	54871	56568	1698	positive	ATG
ATP8a1 genomic sequence	92920	93231	312	positive	ATG
ATP8a1 genomic sequence	104521	104826	306	positive	ATG
ATP8a1 genomic sequence	136402	136773	372	positive	ATG
ATP8a1 genomic sequence	139531	139953	423	positive	ATG
ATP8a1 genomic sequence	152548	152871	324	positive	ATG
ATP8a1 genomic sequence	186019	186384	366	positive	ATG
ATP8a1 genomic sequence	7226	7582	357	positive	ATG
ATP8a1 genomic sequence	32537	32857	321	positive	ATG
ATP8a1 genomic sequence	54902	56518	1617	positive	ATG
ATP8a1 genomic sequence	76304	76642	339	positive	ATG
ATP8a1 genomic sequence	102089	102427	339	positive	ATG
ATP8a1 genomic sequence	169274	169849	576	positive	ATG
ATP8a1 genomic sequence	127864	128187	324	negative	ATG

Table Settings: Column width: Automatic. Show column: Sequence, Start, End, Length, Found at strand, Start codon. Buttons: Select All, Deselect All.

Figure 3.25: A table showing the results of an open reading frames analysis.

Table viewing options in the Side Panel

Options relevant to the view of the table can be configured in the **Side Panel** on the right.

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 be adjusted manually. When the table content exceeds the size of the viewing area, a horizontal scroll becomes available for navigation across the columns.

You can choose which columns can be displayed in the table by checking them in or out from the Side Panel. In some tables, a single checkbox can be used to hide or show a whole set of columns belonging to a certain category. Two buttons called **Select all** and **Deselect all** allow you to select or deselect all columns from that Side Panel section in one click.

Finally, in some table (such as Expression Browser), the content of some columns can be modified using the settings from the Side Panel (such as Expression values and Grouping in figure 3.26).

The screenshot shows the 'Expression Browser' window with a table of 17,490 rows. The table is divided into two main sections: 'DGRP-320' and 'DGRP-370'. Each section has columns for 'SRR' IDs, 'Total counts', and 'Mean'. The 'Mean' column is currently selected for sorting, indicated by a downward arrow. The table lists various genes and their expression values across different SRR runs. For example, 'Ca-P60A' has a mean expression of 5,915.75. The right-hand side of the interface shows the 'Expression Browser Table Settings' panel, where 'Column width' is set to 'Manual', 'Feature information' includes 'Name' and 'Identifier', and 'Expression values' is set to 'Total counts'.

Name	Identifier	DGRP-320				Mean	DGRP-370			
		SRR1543532...	SRR1543624...	SRR1543751...	SRR1543802...		SRR1543519...	SRR1543733...	Mean	SRR
Ca-P60A	FBgn0263006	9,025.00	5,957.00	3,083.00	5,198.00	5,915.75	5,977.00	4,000.00	4,988.50	
CG34220	FBgn0085249	2,032.00	6,835.00	3,738.00	2,238.00	3,710.75	10,081.00	10,858.00	10,469.50	
EF1alpha48D	FBgn0000556	5,562.00	2,468.00	2,182.00	4,385.00	3,649.25	5,286.00	2,550.00	3,918.00	
Peb	FBgn0004181	4,709.00	1,116.00	909.00	4,061.00	2,698.75	4,429.00	920.00	2,674.50	
loopin-1	FBgn0259295	2,381.00	1,043.00	556.00	2,732.00	1,678.00	2,906.00	1,020.00	1,963.00	
Vha16-1	FBgn0262236	1,848.00	1,554.00	898.00	2,279.00	1,644.75	1,951.00	1,482.00	1,716.50	
blw	FBgn0011211	2,387.00	1,292.00	943.00	1,907.00	1,632.25	1,302.00	797.00	1,049.50	
pAbp	FBgn0265297	1,698.00	976.00	717.00	2,344.00	1,433.75	1,818.00	928.00	1,373.00	
CG2127	FBgn0033286	1,757.00	756.00	435.00	2,157.00	1,276.25	2,200.00	812.00	1,506.00	
S-Lap7	FBgn0033868	1,922.00	712.00	419.00	2,038.00	1,272.75	2,539.00	604.00	1,571.50	
shot	FBgn0013733	1,281.00	1,543.00	939.00	1,265.00	1,257.00	1,068.00	1,441.00	1,254.50	
CG8701	FBgn0033287	1,665.00	513.00	323.00	2,107.00	1,152.00	1,995.00	528.00	1,261.50	
Cam	FBgn0000253	1,662.00	753.00	521.00	1,568.00	1,126.00	1,526.00	664.00	1,095.00	
Zasp52	FBgn0265991	1,550.00	1,191.00	615.00	1,078.00	1,108.50	943.00	748.00	845.50	
CG12699	FBgn0046294	1,620.00	560.00	311.00	1,904.00	1,098.75	1,947.00	622.00	1,284.50	
ACC	FBgn0033246	1,145.00	1,272.00	965.00	954.00	1,084.00	784.00	796.00	790.00	
SIP2	FBgn0061197	1,371.00	733.00	440.00	1,786.00	1,082.50	1,639.00	751.00	1,195.00	
Act57B	FBgn0000044	1,562.00	840.00	484.00	1,252.00	1,034.50	1,206.00	641.00	923.50	
CG9975	FBgn0034435	1,226.00	908.00	359.00	1,578.00	1,017.75	1,408.00	770.00	1,089.00	
CG5089	FBgn0034144	1,404.00	557.00	353.00	1,657.00	992.75	1,888.00	541.00	1,214.50	
exu	FBgn0000615	1,319.00	591.00	361.00	1,590.00	965.25	1,667.00	544.00	1,105.50	
CG12860	FBgn0033954	1,346.00	455.00	250.00	1,708.00	939.75	1,886.00	439.00	1,162.50	

Figure 3.26: A table showing the results of an RNA-seq analysis.

Sorting tables

You can **sort** table according to the values of a particular column by clicking a column header. Clicking once will sort in ascending order. A second click will change the order to descending. A third click will set the order back its original order.

Pressing **Ctrl - ⌘** on Mac - while you click other columns will refine the existing sorting with the values of the additional columns, in the order in which you clicked them.

3.3.1 Filtering tables

The final concept to introduce is **Filtering**.

A button called **Filter to selection** allows for reducing the size of a table to a few pre-selected rows. The option **Filter to selected rows** will keep in the table view only the rows that are selected, whether they were selected manually, or by using the function "Select in other views" available for some tables (for example when the table is associated with a graphical view such as a Venn diagram, or a volcano plot). Restore the complete table by choosing the option **Clear selection filter**.

In the top right corner of a table view, a filter field can be used to filter a table as well. This table filter has 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 3.27).

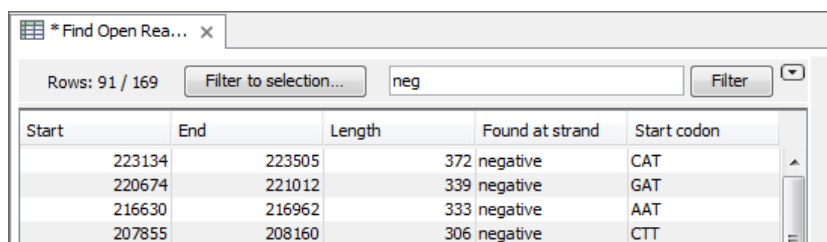


Figure 3.27: Typing "neg" in the filter in simple mode.

Typing "neg" in the filter will only show the rows where "neg" is part of the text in any of the columns. 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. Filtering is automatic once you start typing, unless you are working with a table with more than 10000 rows, in which case you have to actually click the **Filter** button for the filtering to take effect.

In the advanced mode, you can make use of numerical information or make more complex filter combinations using more than one criterion in the filter. Click the **Advanced filter** (▼) button to open the first criterion of the advanced 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)

Note, that the number of digits displayed is a formatting option which can be set in the View Preferences. The true number may well be (slightly) larger. This behaviour can lead to problems when filtering on exact matches using the = (equal to) operator on numbers. Instead, users are advised to use two filters of inequalities (< (smaller than) and > (greater than)) delimiting a (small) interval around the target value.

For text-based columns, you can choose between:

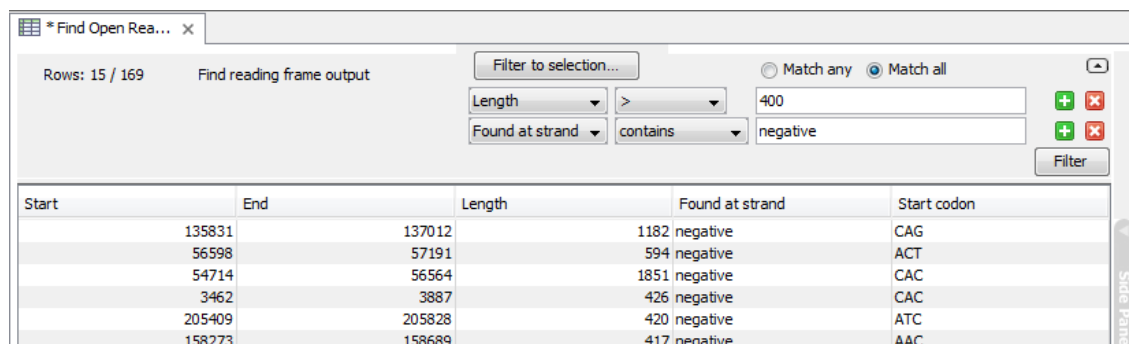
- **starts with** (the text starts with your search term)
- **contains** (the text does not have to be in the beginning)
- **doesn't contain**
- **=** (the whole text in the table cell has to match, also lower/upper case)
- **≠** (the text in the table cell has to not match)
- **is in list** (The text in the table cell has to match one of the items of the list. Items are separated by comma, semicolon, or space. This filter is not case-sensitive.)
- **is not in list** (The text in the table cell must not match any of the items of the list. Items are separated by comma, semicolon, or space. This filter is not case-sensitive)

Once you have chosen an operator, you can enter the **text or numerical value** to use.

The advanced filter criterion mentioned above are also available from a menu that appears by right-clicking on a value in a table: just specify the operator, and the column and value where you right-clicked for the menu to appear will define the two other fields of the advanced filter.

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 3.28 shows an example of an advanced filter which displays the open reading frames larger than 400 that are placed on the negative strand.



Start	End	Length	Found at strand	Start codon
135831	137012	1182	negative	CAG
56598	57191	594	negative	ACT
54714	56564	1851	negative	CAC
3462	3887	426	negative	CAC
205409	205828	420	negative	ATC
158273	158689	417	negative	AAC

Figure 3.28: The advanced filter showing open reading frames larger than 400 that are placed on the negative strand.

Both for the simple and the advanced filter, there is a counter at the upper left corner which tells you the number of rows that pass the filter (91 in figure 3.27 and 15 in figure 3.28).

3.4 Customized attributes on data locations

Location-specific attributes can be set on all elements stored in a given data location. Attributes could be things like company-specific information such as LIMS id, freezer position etc. Attributes are set using a CLC Workbench acting as a client to the CLC Server.

Note that the attributes scheme belongs to a particular data location, so if there are multiple data locations, each will have its own set of attributes.

To configure which fields that should be available¹ go to the Workbench:

right-click the data location | Location | Attribute Manager

This will display the dialog shown in figure 3.29.

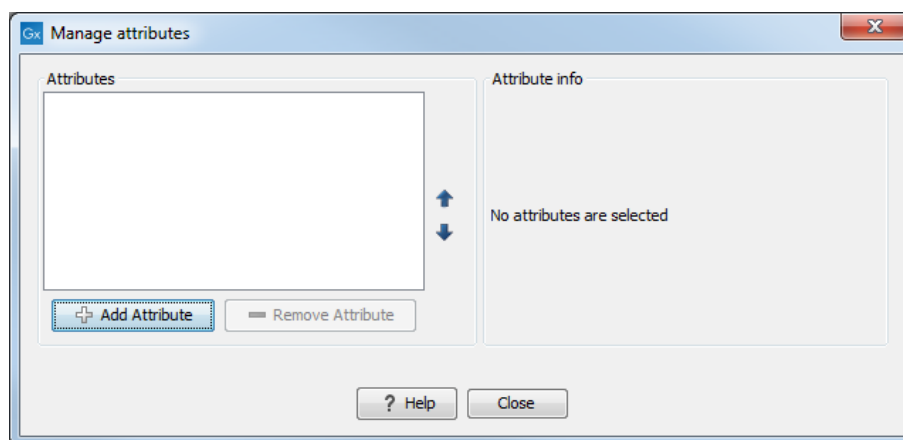


Figure 3.29: Adding attributes.

Click the **Add Attribute** (+) button to create a new attribute. This will display the dialog shown in figure 3.30.

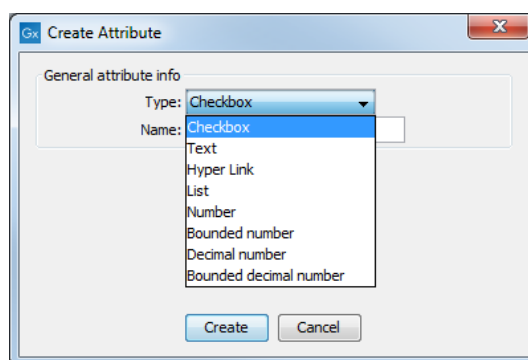


Figure 3.30: The list of attribute types.

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.
- **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).

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

- **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.

Lists are a little special, since you have to define the items in the list. When you choose to add the list attribute 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 3.31).

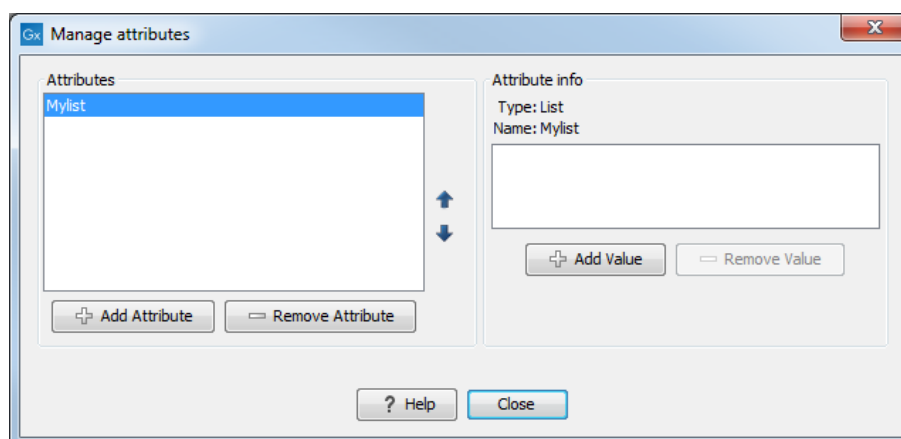


Figure 3.31: Defining items in a list.

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

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.

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.

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.

3.4.1 Filling in values

When a set of attributes has been created (as shown in figure 3.32), the end users can start filling in information.

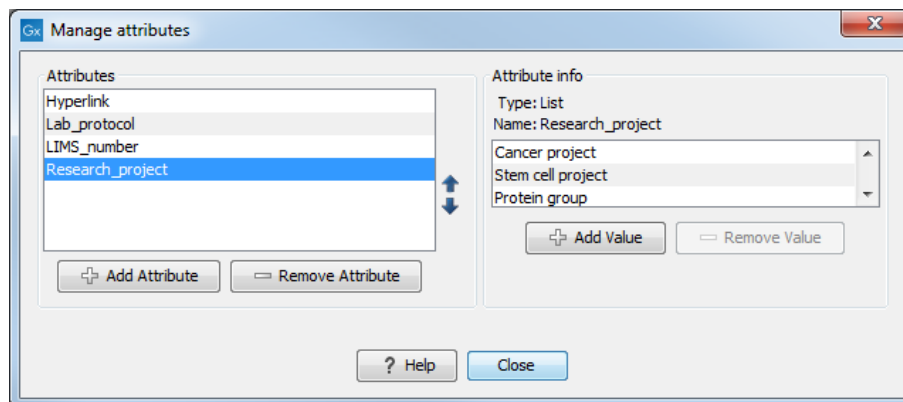


Figure 3.32: A set of attributes defined in the attribute manager.

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 3.33.

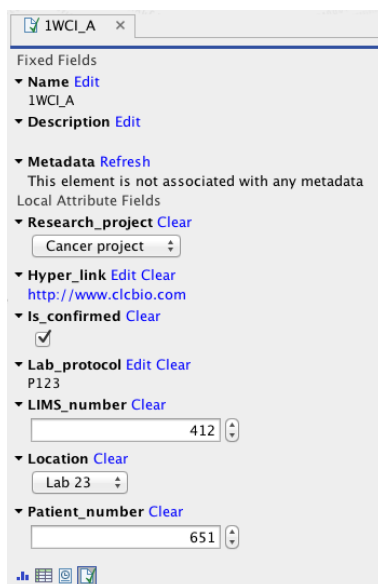


Figure 3.33: Adding values to the attributes.

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 element (e.g. 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 3.34).

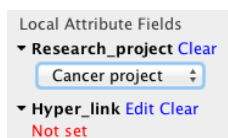


Figure 3.34: An attribute which has not been set.

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 3.34, 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**, invoked by pressing **Show** on a given folder from the context menu, provides a quick way of changing the attributes of many elements in one go (see section 3.1.8).

3.4.2 What happens when a clc object 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 element (e.g. sequence) is moved back to the original data location, the information will again be editable and searchable.

If the e.g. Molecule Project or Molecule Table is moved back to the original data location, the information will again be editable and searchable.

3.4.3 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 3.35).

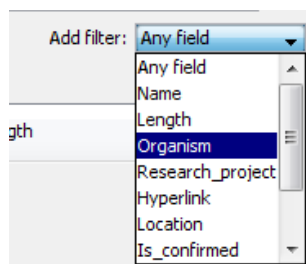


Figure 3.35: The attributes from figure 3.32 are now listed in the search filter.

It will also be available in the **Quick Search** below the **Navigation Area** (press Shift+F1 (Fn+Shift+F1 on Mac) and it will be listed - see figure 3.36).

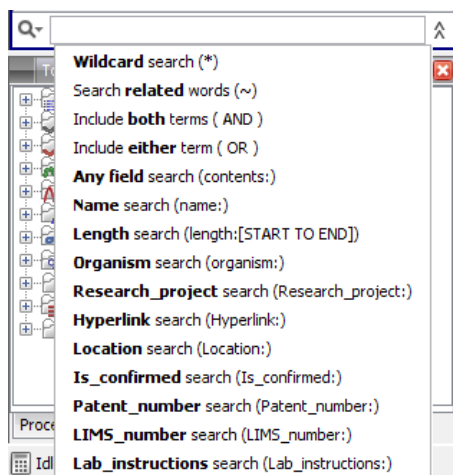


Figure 3.36: The attributes from figure 3.32 are now available in the Quick Search as well.

Read more about search here: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Local_search.html.

3.5 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.

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 3.4

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 23.1.2) instead.

Search index This section has a technical focus and is not relevant if your searches are working well.

However, if you experience problems with your search results, i.e., 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 2.3.1.

3.5.1 Quick search

At the bottom of the **Navigation Area** there is a text field as shown in figure 3.37). To search, simply enter a text to search for and press **Enter**.

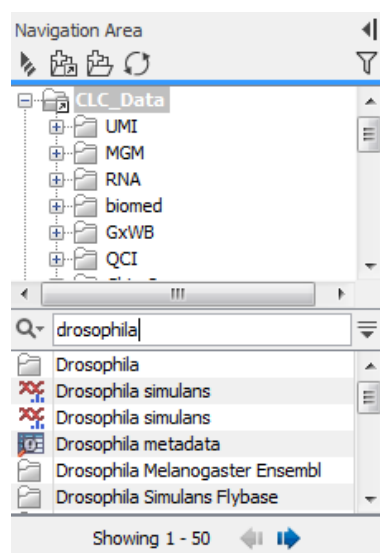


Figure 3.37: Search simply by typing in the text field and press Enter.

Note that the search term supports advanced features known from web search engines, which means that the following list of characters carry special meaning: + - & | ! () ^ [] " ~ * ? : \ / . To avoid this special interpretation it is suggested to put quotes around the search expression when searching for data containing the special characters, or read the section 3.5.2 on advanced search expressions.

To show the results, the search pane is expanded.

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. In the preferences (see Chapter 4), you can specify the number of hits to be shown.

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 highlight the search hit in its folder in the **Navigation Area**.

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 3.38. 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.

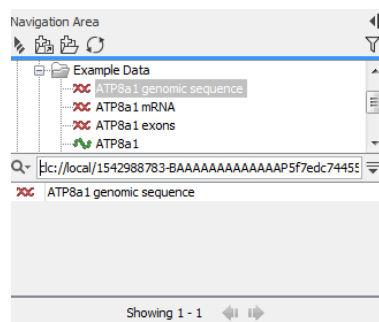


Figure 3.38: The search field can also be used to search for data locations.

Quick search history You can access the 10 most recent searches by clicking the icon (Q) next to the search field (see figure 3.39).

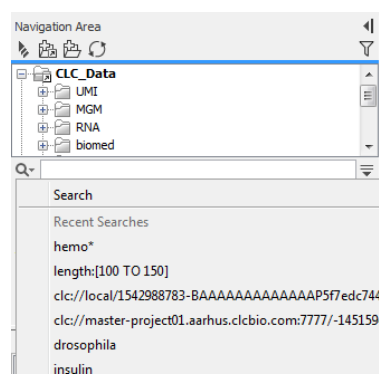


Figure 3.39: Recent searches.

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

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 3.40.

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:

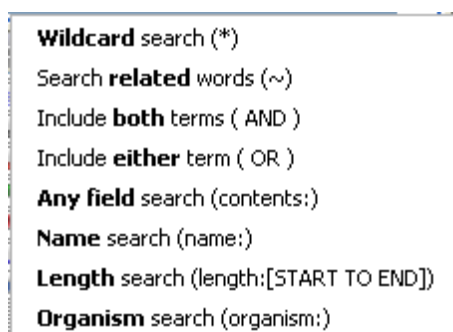


Figure 3.40: Guides to help create advanced search expressions.

- **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 tilde 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 11.4).
- **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 3.4), 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.

3.5.2 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)

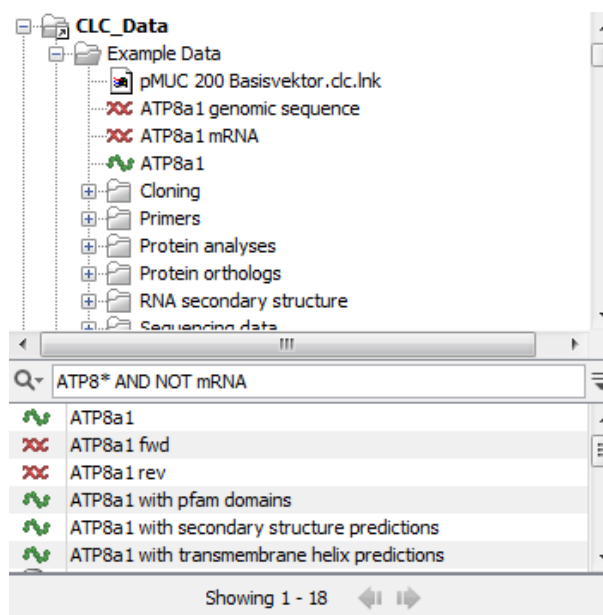


Figure 3.41: An example of searching for elements with the name, description and organism information that includes "ATP8" but do not include the term "mRNA".

The first thing you can choose is which location should be searched. All the active locations are shown in this list. You can also choose to search all locations. Read more about locations in section 3.1.1.

Furthermore, you can specify what kind of elements should be searched:

- 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 3.5.1 for more information on individual search terms.

For all other data, you can only search for name.

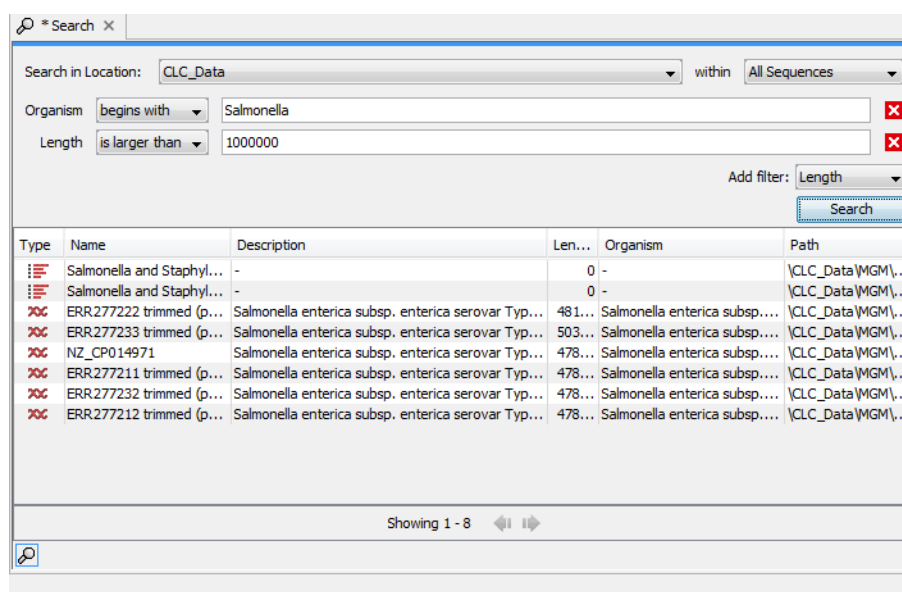
If you use **Any field**, it will search all of the above plus the following:

- Description
- Keywords
- Common name
- Taxonomy name

To see this information for a sequence, switch to the **Element Info** () view (see section 11.4).

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 3.42.



The screenshot shows a search window titled "* Search". The search criteria are:


- Search in Location: CLC_Data
- within: All Sequences
- Organism: begins with Salmonella
- Length: is larger than 1000000
- Add filter: Length

 The search results are displayed in a table with the following columns: Type, Name, Description, Len..., Organism, and Path. The results show several entries for Salmonella sequences, including trimmed reads and specific accession numbers like NZ_CP014971. The table indicates that 8 results are shown out of a total of 8.

Type	Name	Description	Len...	Organism	Path
	Salmonella and Staphyl...	-	0 -	-	\CLC_Data\MGM\...
	Salmonella and Staphyl...	-	0 -	-	\CLC_Data\MGM\...
	ERR277222 trimmed (p...	Salmonella enterica subsp. enterica serovar Typ...	481...	Salmonella enterica subsp....	\CLC_Data\MGM\...
	ERR277233 trimmed (p...	Salmonella enterica subsp. enterica serovar Typ...	503...	Salmonella enterica subsp....	\CLC_Data\MGM\...
	NZ_CP014971	Salmonella enterica subsp. enterica serovar Typ...	478...	Salmonella enterica subsp....	\CLC_Data\MGM\...
	ERR277211 trimmed (p...	Salmonella enterica subsp. enterica serovar Typ...	478...	Salmonella enterica subsp....	\CLC_Data\MGM\...
	ERR277232 trimmed (p...	Salmonella enterica subsp. enterica serovar Typ...	478...	Salmonella enterica subsp....	\CLC_Data\MGM\...
	ERR277212 trimmed (p...	Salmonella enterica subsp. enterica serovar Typ...	478...	Salmonella enterica subsp....	\CLC_Data\MGM\...

Figure 3.42: Searching for Salmonella sequences larger than 1 million 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.

Chapter 4

User preferences and settings

Contents

4.1	General preferences	98
4.2	View preferences	100
4.2.1	Import and export Side Panel settings	101
4.3	Data preferences	103
4.4	Advanced preferences	103
4.5	Export/import of preferences	104
4.6	View settings for the Side Panel	105

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 4.1:

Edit | Preferences (⚙️)

or **Ctrl + K** (⌘ + ; on Mac)

4.1 General preferences

The **General preferences** include:

- **Undo Limit.** As default the undo limit is set to 500. By writing a higher number in this field, more actions can be undone. Undo applies to all changes made on molecules, sequences, alignments or trees (see section 2.1.5).
- **Audit Support.** If this option is checked, all manual editing of sequences will be marked with an annotation on the sequence (see figure 4.2). Placing the mouse on the annotation will reveal additional details about the change made to the sequence (see figure 4.3). Note

- Errors (but without any information that could lead to loss of privacy: file names and organisms will not be logged)
- Installation and removal of plugins and modules

The following information is also sent:

- An installation ID. This allows us to group events coming from the same installation. It is not possible to connect this ID to personal or license information.
- A geographic location. This is predicted based on the IP-address. We do not store IP-addresses after location information has been extracted.
- A time stamp

4.2 View preferences

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

1. **Toolbar** lets you choose the size of the toolbar icons, and whether to display names below the icons (figure 4.4).

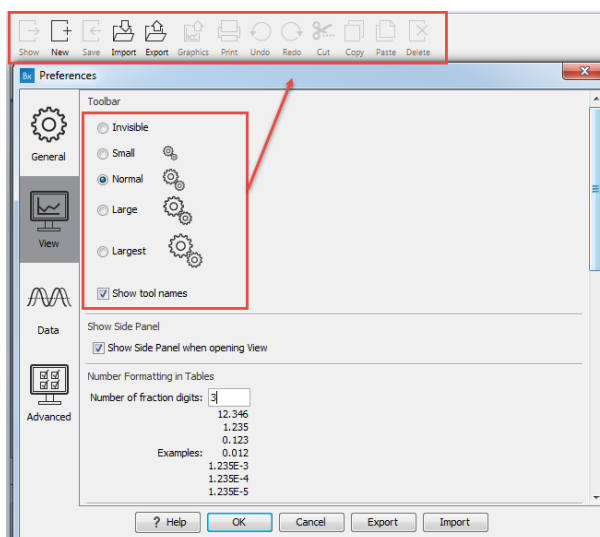


Figure 4.4: Number formatting of tables.

2. **Show Side Panel** allows you to choose whether to display the side panel when opening a new view. Note that for any open view, the side panel can be collapsed by clicking on the small triangle at the top left side of the settings area or by using the key combination Ctrl + U (⌘ + U on Mac).
3. **Number formatting in tables** specifies how the numbers should be formatted in tables (see figure 4.5). 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.
4. **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).

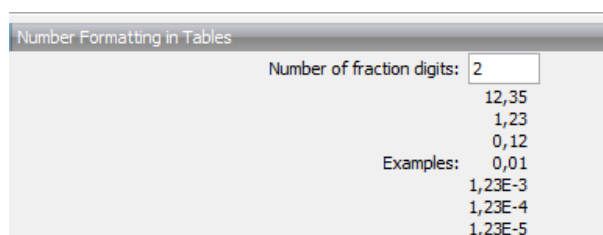


Figure 4.5: Number formatting of tables.

- Accession (sequences downloaded from databases like GenBank have an accession number).
 - Latin name.
 - Latin name (accession).
 - Common name.
 - Common name (accession).
5. **User Defined View Settings** gives you an overview of the different Side Panel settings that are saved for each view. See section 4.6 to learn 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 4.6).

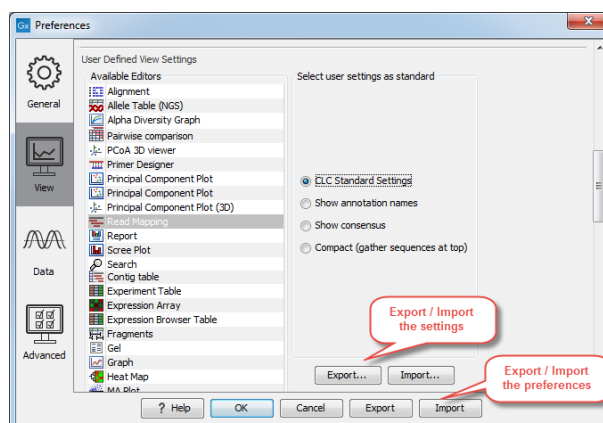


Figure 4.6: Selecting the default view setting.

Note that the content of this list depends on the nature of the elements that are saved in the Navigation Area. When the list grows, you may have to scroll up or down to find the relevant settings.

6. **Molecule Project 3D Editor** gives you the option to turn off the modern OpenGL rendering for **Molecule Projects** (see section 12.2).

4.2.1 Import and export Side Panel settings

If you have created a special set of settings in the **Side Panel** that you wish to share with other CLC users, you can export the settings in a file. The other user can then import the settings.

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

that is situated below the list of possible settings (see figure 4.6), and not the Export button at the very bottom of the dialog, as this one will export the **Preferences** (see section 4.5).

A dialog will be shown (see figure 4.7) that allows you to select which of the settings you wish to export.

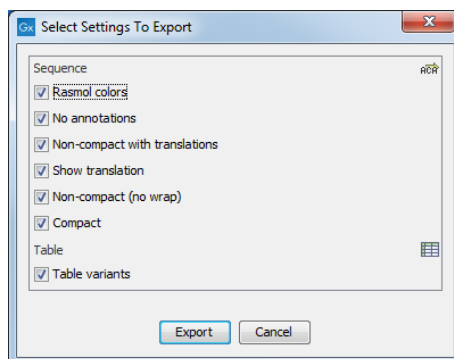


Figure 4.7: Exporting all settings for circular views.

When multiple views are selected for export, all the view settings for the views will be shown in the dialog. Click **Export** and you will now be able to define a save folder and name for the exported file. The settings are saved in a file with a .vsf extension (View Settings File).

Similarly, to import a **Side Panel** settings file, make sure you are at the bottom of the **View** panel of the **Preferences dialog**, and click the **Import...** button. Note that there is also another import button at the very bottom of the dialog, but this will import the other settings of the **Preferences** dialog (see section 4.5).

Select the *.vsf file where the settings are saved. The following 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 4.8).

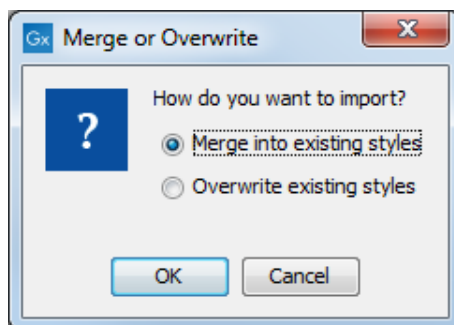


Figure 4.8: When you import settings, you are asked if you wish to overwrite existing settings or if you wish to merge the new settings into the old ones.

WARNING! If you choose to overwrite the existing settings, you will lose ALL the Side Panel settings that were 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 6.1).
- **Graphics** export of the views which creates image files in various formats (described in

section 6.3).

- 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.

4.3 Data preferences

The data preferences contain preferences related to interpretation of data:

- Multisite Gateway Cloning primer additions, a list of predefined primer additions for Gateway cloning (see section 20.4.1).

4.4 Advanced preferences

Proxy Settings The Advanced settings include the possibility to set up a proxy server. This is described in section 1.6.

Default data location The default location is used when you import a file without selecting a folder or element in the Navigation Area first. It is set to the folder called CLC_Data in the Navigation Area, but can be changed to another data location using a drop down list of data locations already added (see section 3.1.1). Note that the default location cannot be removed, but only changed to another location.

Data Compression CLC format data is stored in an internally compressed format. The application of internal compression can be disabled by unchecking the option "Save CLC data elements in a compressed format". This option is enabled by default. Turning this option off means that data created may be larger than it otherwise would be.

Enabling data compression may impose a performance penalty depending on the characteristics of the hardware used. However, this penalty is typically small, and we generally recommend that this option remains enabled.

Internal compression of CLC data was introduced in CLC Genomics Workbench 12.0, CLC Main Workbench 8.1 and CLC Genomics Server 11.0. CLC format files with internal compression are not compatible with older versions of the software. Turning this option off is likely to be of interest only at sites running a mix of older and newer CLC software, where the same data is accessed by different versions of the software.

To work with specific data sets in older CLC software versions, we recommend exporting the data to CLC or zip format and turning on the export option "Maximize compatibility with older CLC products". This is described in more detail in section 6.2.4.

NCBI Integration Without an API key, access to NCBI from a single IP-address is limited to 3 requests per second; if many workbenches use the same IP address when running the Search for Reads in SRA..., Search for Sequences at NCBI, and Search for PDB Structures at NCBI tools

they may hit this limit. In this case, you can create an API key for NCBI E-utilities in your NCBI account and enter it here.

NCBI BLAST The standard URL for the BLAST server at NCBI is: <https://blast.ncbi.nlm.nih.gov/Blast.cgi>, but it is possible to specify an alternate server URL to use for BLAST searches. Be careful to specify a valid URL, otherwise BLAST will not work.

4.5 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 and click on the Export button at the bottom of the Preferences dialog. Select the relevant preferences and click Export to choose a location to save the exported file (see figure 4.9).

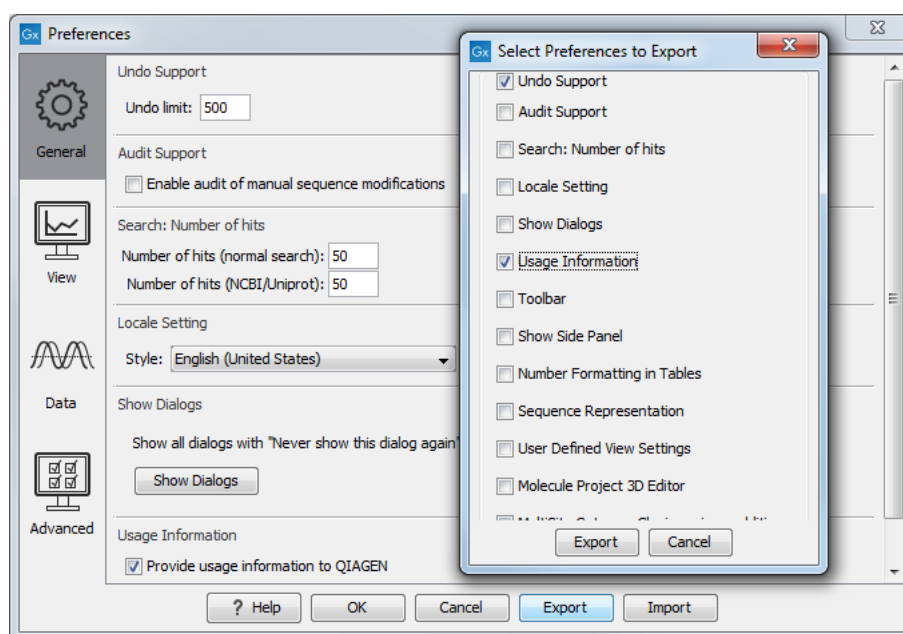


Figure 4.9: Select which of the preferences you want to export.

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 4.2.1.

The process of importing preferences is similar to exporting: click the Import button and browse to the *.cpf file.

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 6.1).

- **Graphics** export of the views that create image files in various formats (described in section 6.3).
- 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.

4.6 View settings for the Side Panel

The **Side Panel** is shown to the right of all views that are opened in the View Area. 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 (see section 2.1.8).

The options for saving and applying are available at the bottom of the **Side Panel** (see figure 4.10).

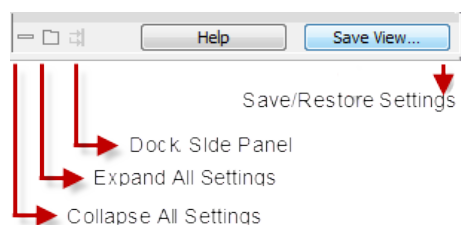


Figure 4.10: Functionalities found at the bottom of the Side Panel.

Opening a view type (e.g., a circular sequence, a variant table, or a PCA) for the first time will display the element using the CLC Standard Settings for that type of view. You can then adjust the settings using all the options available to you in the side panel. When you have adjusted a view to your preference, the new settings can be saved (see figure 4.11).

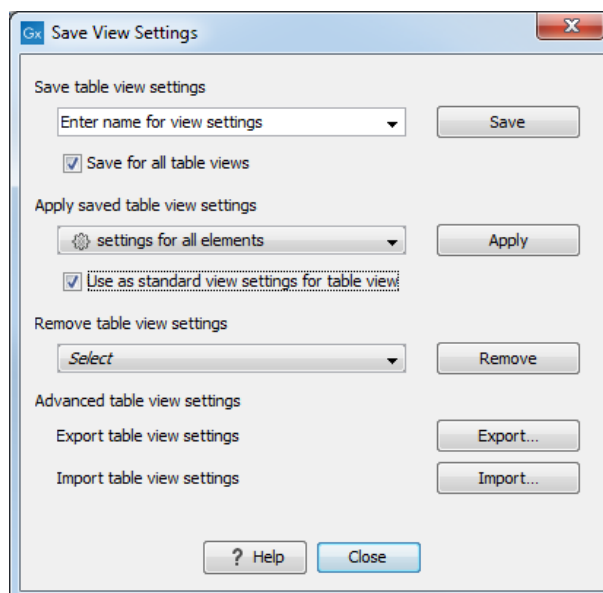


Figure 4.11: Functionalities found at the bottom of the Side Panel.

Saving can be done two ways. Write a name for the particular settings you just set, and choose to save:

- For that view alone, so that the settings will be available to you the next time you open this particular element. 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.
- For all other views, when the option "Save for all element views" is checked, so that the settings will be available to you the next time you open any element for which this type of view is available.

Similarly, applying can be done two ways:

- For that view alone, so that the settings are applied the next time you open this particular element.
- For all other elements, when the option "Use as standard view settings for element view" is checked, so that the settings are applied each time you open any element for which this type of view is available. These "general" settings are user specific and will not be saved with or exported with the element.

"General" settings can be shared and imported with other workbench users using the **Export** and **Import** buttons at the bottom of the dialog. Exporting and importing saved settings can also be done in the **Preferences** dialog under the **View** tab (see section 4.2.1).

It is possible to remove a saved setting using the saved settings list from the drop-down menu and clicking **Remove**.

Chapter 5

Printing

Contents

5.1	Selecting which part of the view to print	108
5.2	Page setup	109
5.3	Print preview	111

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 6.3) 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 5.1).

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.

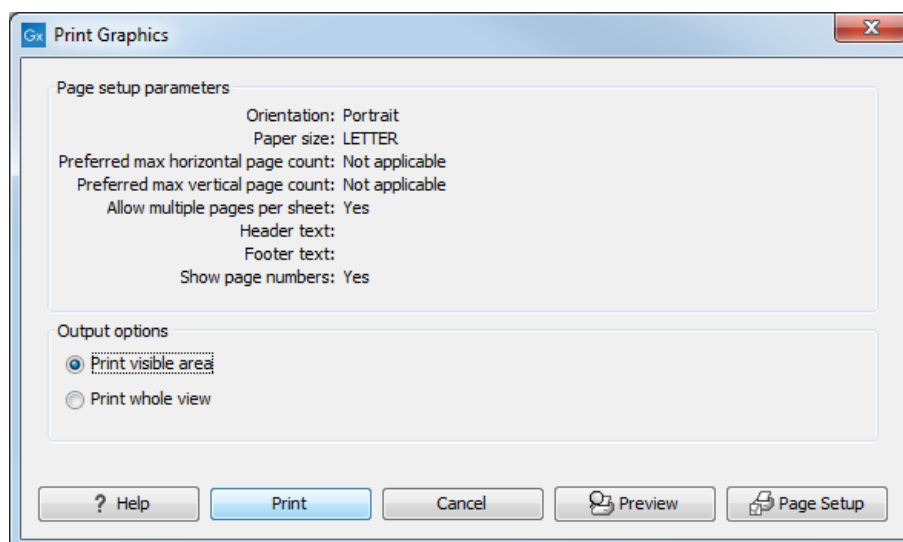


Figure 5.1: The Print dialog.

5.1 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 5.2 is a view of a circular sequence which is zoomed in so that you can only see a part of it.

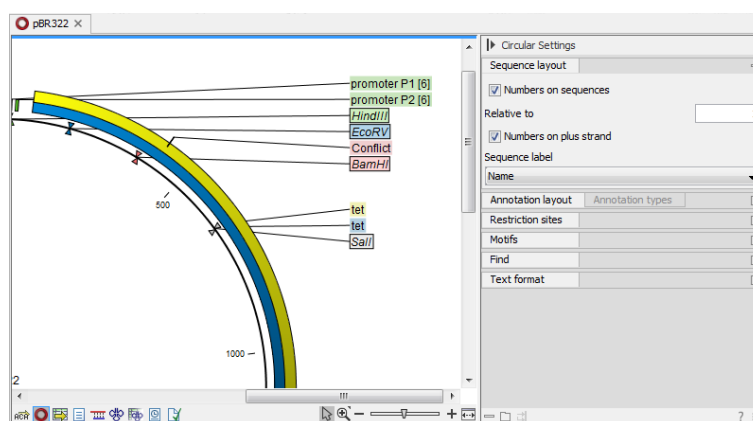


Figure 5.2: A circular sequence as it looks on the screen.

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 5.2 and choosing **Print visible area** can be seen in figure 5.3.

On the other hand, if you select **Print whole view**, you will get a result that looks like figure 5.4. This means that you also print the part of the sequence which is not visible when you have zoomed in.

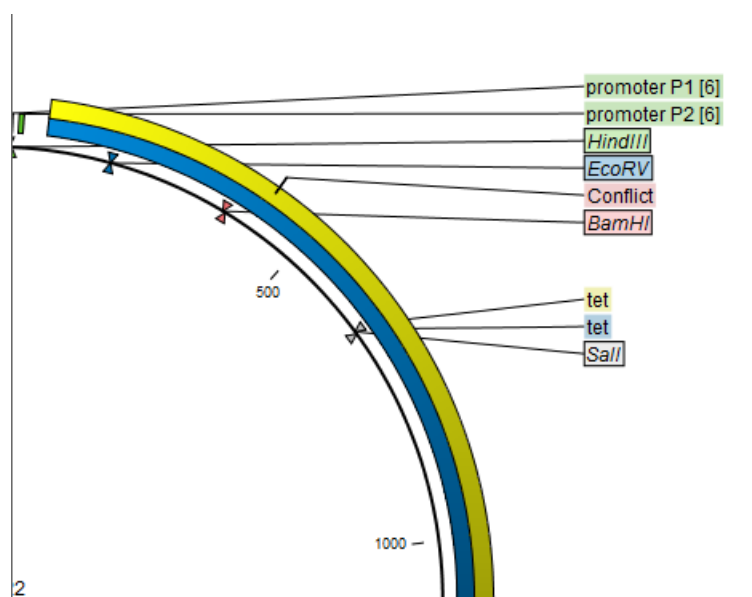


Figure 5.3: A print of the sequence selecting Print visible area.

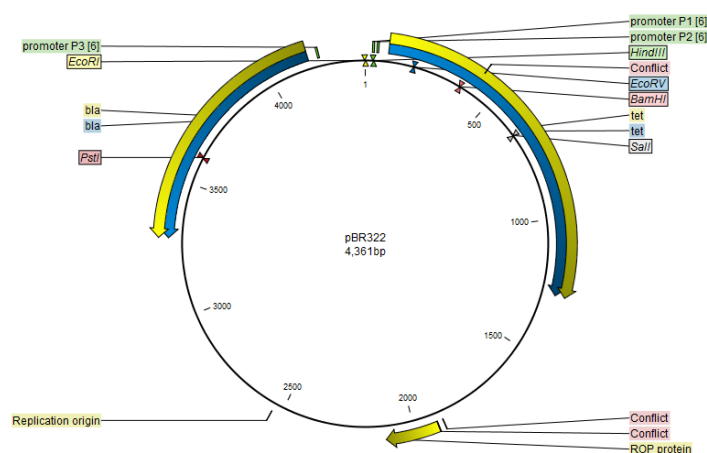


Figure 5.4: A print of the sequence selecting Print whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.

5.2 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 5.5

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.

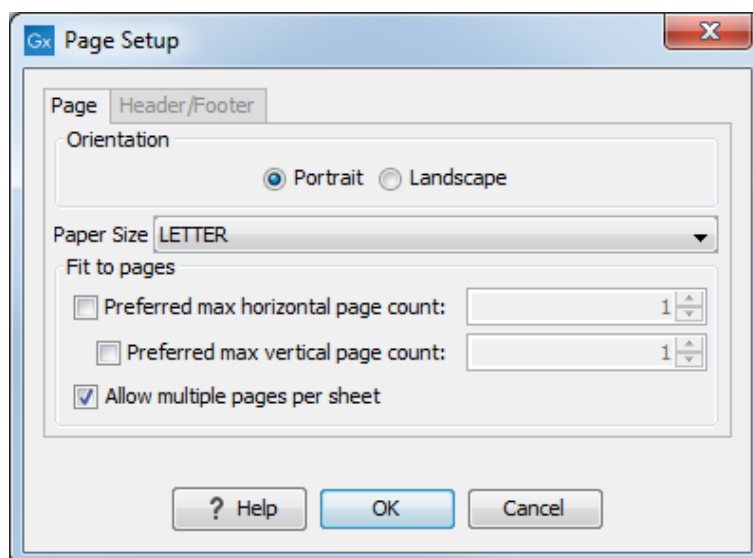


Figure 5.5: Page Setup.

- **Fit to pages.** Can be used to control how the graphics should be split across pages (see figure 5.6 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.

Figure 5.6: An example where *Fit to pages horizontally* is set to 2, and *Fit to pages vertically* is set to 3.

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**.

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.

5.3 Print preview

The preview is shown in figure 5.7.

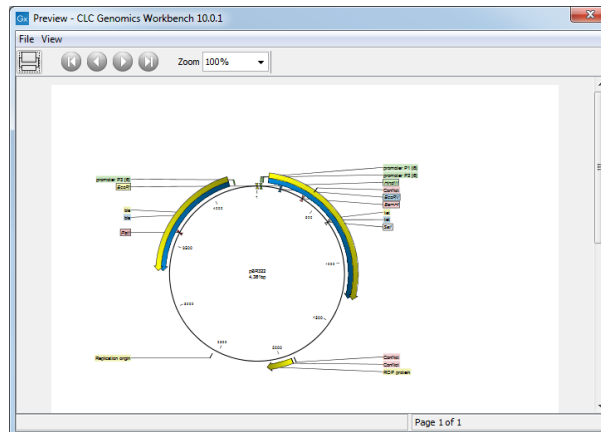



Figure 5.7: *Print preview.*

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 6

Import/export of data and graphics

Contents

6.1 Standard import	112
6.1.1 External files	114
6.2 Data export	114
6.2.1 Export formats	115
6.2.2 Export parameters	116
6.2.3 Choosing the exported file name(s)	117
6.2.4 Export of folders and data elements in CLC format	119
6.2.5 Export of dependent elements	120
6.2.6 Export history	121
6.2.7 Backing up data from the CLC Workbench	122
6.2.8 Export of tables	123
6.3 Export graphics to files	124
6.3.1 File formats	127
6.4 Export graph data points to a file	129
6.5 CLC Server data import and export	130
6.6 Copy/paste view output	130

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 (). 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.

6.1 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 [H.1](#).

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

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 6.1. You can change which kind of file types that should be shown by selecting a file format in the **Files of type** box.

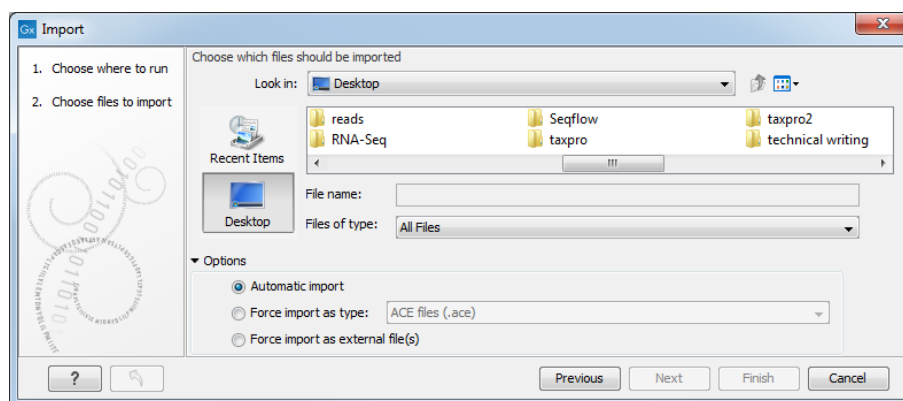


Figure 6.1: The import dialog.

Next, select one or more files or folders to import and click **Next** 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 6.1), 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.

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.

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.

6.1.1 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 6.1). Bioinformatics files not recognized by *CLC Main Workbench* are also treated as external files.

There is a special tool for importing data from Vector NTI. This tool is a plugin which can be downloaded and installed in the *CLC Main Workbench* using the plugin manager (see section 1.5).

6.2 Data export

Data can be exported from the *CLC Main Workbench* to many standard formats. Supported formats are listed in section H.1, but an easy way to see the full list is to launch the Export tool, where they are presented in the first dialog window.

Launch the standard export functionality by clicking on the Export button on the toolbar, or selecting the menu option:

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 6.2.5.

The general steps when configuring a standard export job are:

- (Optional) Select data elements or folders to export in the **Navigation Area**.
- Launch the Export tool by clicking on the Export button in the Workbench toolbar or by selecting **Export** under the File menu.

- Select the format to export the data to.
- Select the data elements to export, or confirm elements that had been pre-selected in the **Navigation Area**.
- Configure the export parameters, including whether to output to a single file, whether to compress the outputs and how the output files should be named. Other format-specific options may also be provided.
- Select where the data should be exported to.
- Click on the button labeled **Finish**.

6.2.1 Export formats

Finding and selecting a format to export to When the Export tool is launched, a list of the available data formats is presented.

You can quickly find a particular format by typing a relevant search term into the text box at the top of the Export window, as shown in figure 6.3. Any formats with that search term in their name or description will be listed in the window. The search term is remembered when the Export tool is next launched. Delete the text from the search box if you wish to have all export formats listed.

Support for choosing an appropriate export format is provided in 2 ways:

- If data elements are selected in the **Navigation Area** before launching the Export tool, then a "Yes" or a "No" in the **Supported formats** column specifies whether or not the selected data elements can be exported to that format. If you have selected multiple data elements of different types, then formats that some, but not all, selected data elements can be exported to are indicated by the text "For some elements".

By default, supported formats appear at the top of the list (figure 6.2).

- If no data elements are selected in the **Navigation Area** when the Export tool is launched, then the list of export formats is provided, but each row will have a "Yes" in the **Supported format** column. After an export format has been selected, only the data elements that can be exported to that format will be listed for selection in the next step of the export process.

Only zip format is supported when a folder, rather than data elements, is selected for export. In this case, all the elements in the folder are exported in CLC format, and a zip file containing these is created. This is described in more detail in section 6.2.4.

When the desired export format has been selected, click on the button labeled **Select**.

A dialog then appears, with a name reflecting the format you have chosen. For example if the 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.

Selecting data for export In figure 6.4 we show the selection of a variant track for export to VCF format.

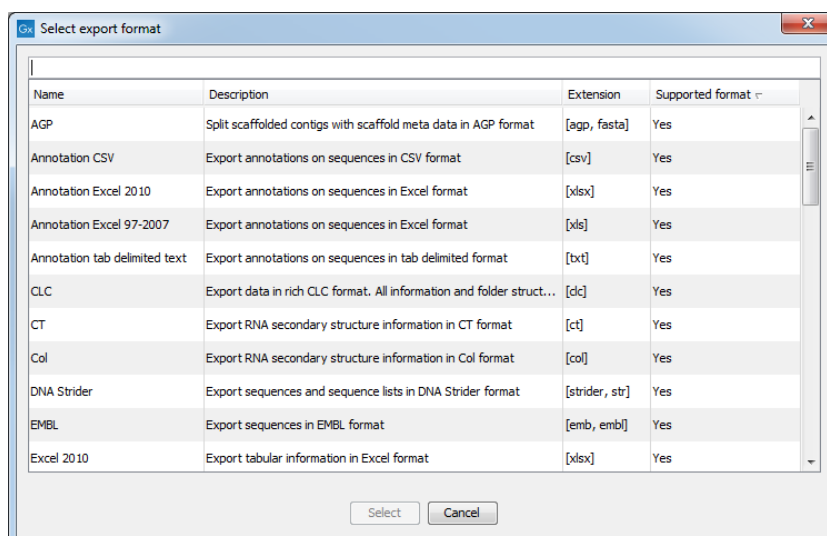


Figure 6.2: The Select export format dialog. Here, some sequence lists had been selected in the Navigation Area before the Export tool was launched. The formats that the selected data elements can be exported to contain a "Yes" in the Selected format column. Other export formats are listed below the supported ones, with "No" in the Supported format column.

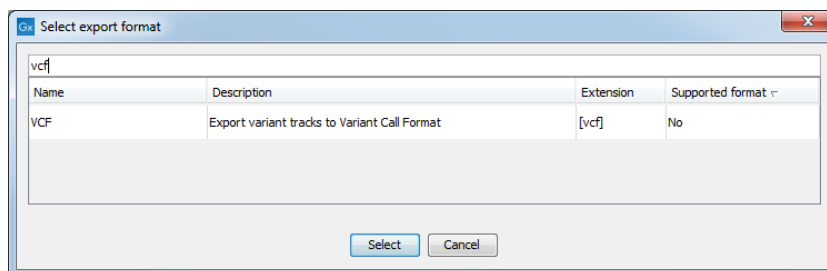


Figure 6.3: The text field has been used to search for the term "VCF" in the export format name or description field in the Select export dialog.

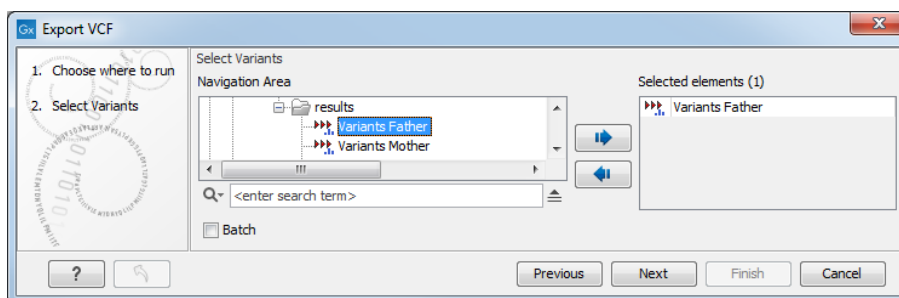


Figure 6.4: The Select export dialog. Select the data element(s) to export.

6.2.2 Export parameters

The settings in the areas **Basic export parameters** and **File name** are offered when exporting to any format.

There may also be additional parameters for particular export formats. This is illustrated for the CLC exporter in figure 6.5.

Examples of configuration options:

- **Compression options.** Within the **Basic export parameters** section, you can choose to

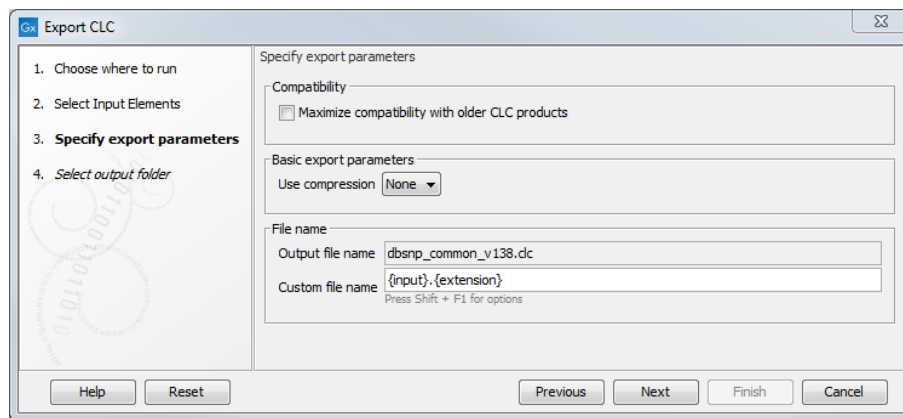


Figure 6.5: Configure the export parameters. When exporting to CLC format, you can choose to maximize compatibility with older CLC products.

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.

- **Paired reads settings.** In the case of Fastq Export, the option "Export paired sequence lists to two files" is selected by default: it will export paired-end reads to two fastq files rather than a single interleaved file.
- **Exporting multiple files.** If you have selected multiple files of the same type, you can choose to export them in one single file (only for certain file formats) by selecting "Output as single file" in the **Basic export parameters** section. If you wish to keep the files separate after export, make sure this box is not ticked. **Note:** Exporting in zip format will export only one zipped file, but the files will be separated again when unzipped.

After configuration, choose where to save the exported files to.

6.2.3 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.

When working with the export of multiple files, using some combination of the terms shown by default in this field and in figure 6.6 are recommended. Clicking in the **Custom file name** field with the mouse and then simultaneously pressing the Shift + F1 keys bring up a list of the available terms that can be included in this field.

The following placeholders are available:

- **{input}** or **{1}** - default name of the data element being exported
- **{extension}** or **{2}** - default extension for the chosen export format
- **{counter}** or **{3}** - a number that is incremented per file exported. i.e. If you export more than one file, counter is replaced with 1 for the first file, 2 for the next and so on.

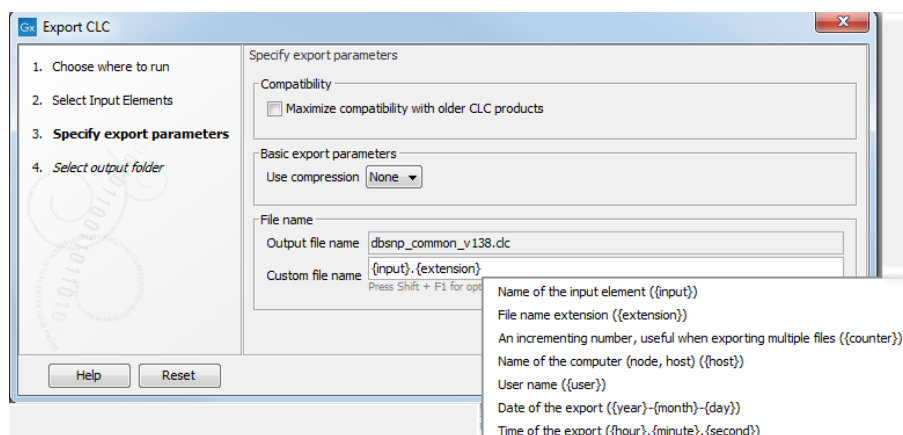


Figure 6.6: Use the custom file name pattern text field to make custom names.

- **{user}** - name of the user who launched the job
- **{host}** - name of the machine the job is run on
- **{year}**, **{month}**, **{day}**, **{hour}**, **{minute}**, and **{second}** - timestamp information based on the time an output is created. Using these placeholders, items generated by a workflow at different times can have different filenames.

In the following example, we would like to change the export file format to .fasta in a situation where .fa was the default format that would be used if you kept the default file extension suggestion ("{2}"). To do this, replace "{2}" with ".fasta" in the "Custom file name field". You can see that when changing "{2}" to ".fasta", the file name extension in the "Output file name" field automatically changes to the new format (see figure 6.7).

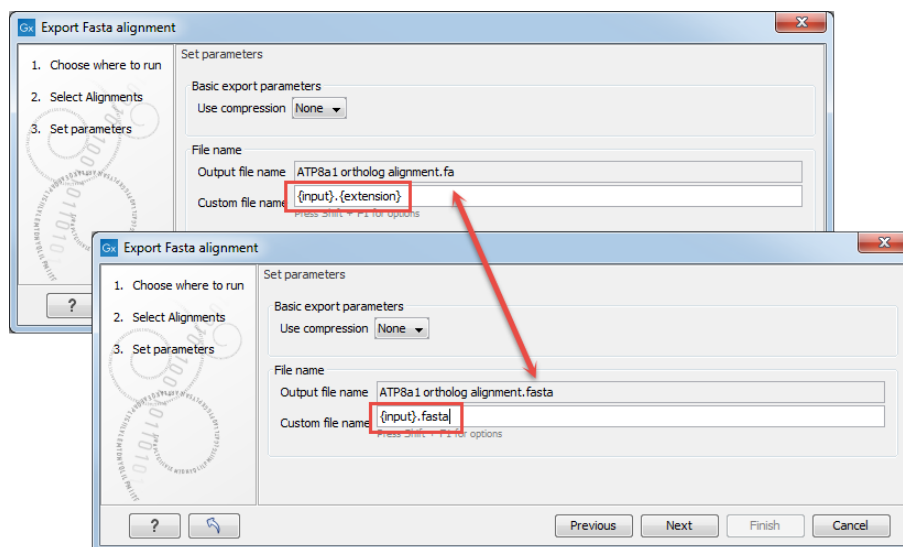


Figure 6.7: The file name extension can be changed by typing in the preferred file name format.

When deciding on an output name, you can choose any combination of the different placeholders as well as custom names and punctuation, as in `{input}({day}-{month}-{year})`.

Another example of a meaningful name to a variant track could be `{2} variant track` as shown in figure 6.8. If your workflow input is named `Sample 1`, the result would be "Sample 1 variant track".

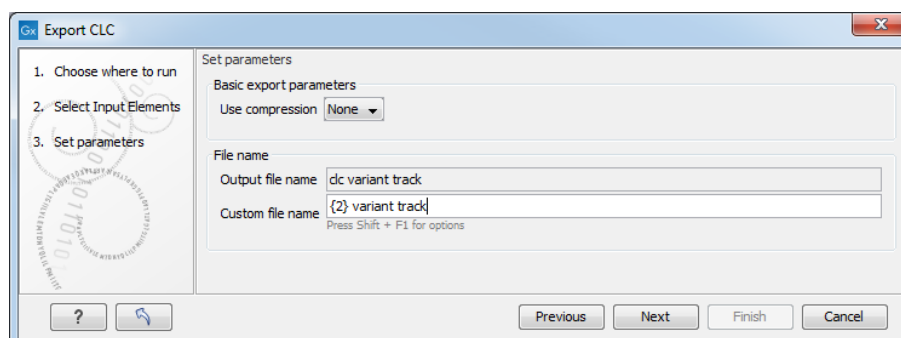


Figure 6.8: Providing a custom name for the result.

As you add or remove text and terms in the **Custom 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.

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 6.9).

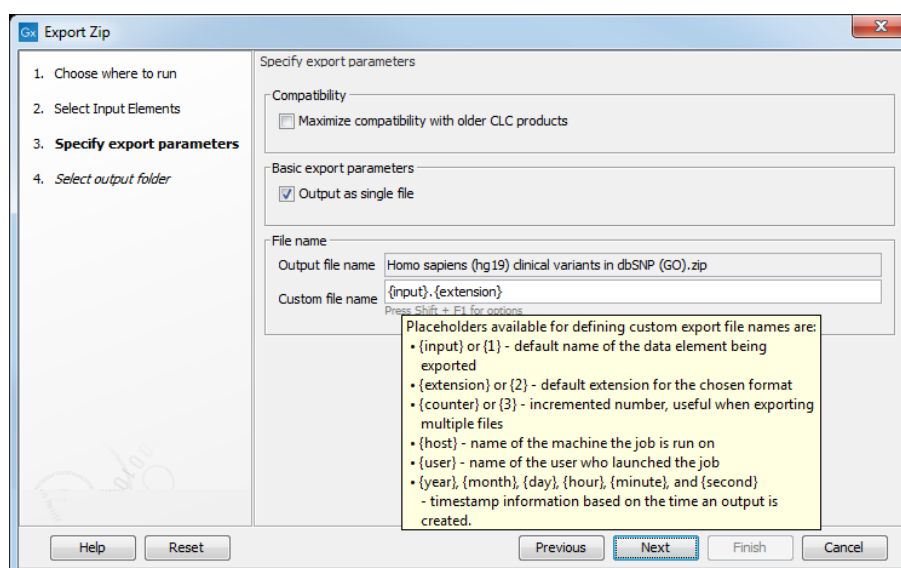


Figure 6.9: The output file names are listed in the "Output file name" text field.

6.2.4 Export of folders and data elements in CLC format

The *CLC Main Workbench* stores data in CLC format. A CLC format file holds all the information for a given data element. This means the data itself, as well as information about that data, like history information.

Data can be exported in CLC format by selecting the CLC format, or the zip format, from the list of available formats.

If CLC format is chosen, each selected data element can be exported to an individual file. An

option is offered later in the export process to apply gzip or zip compression. Choosing gzip compression at this stage will compress each data element individually. Choosing zip produces a single file containing the individual CLC format files. If a single zip file containing one or more CLC format files is the desired outcome, choosing the zip format in the first step of the export process specifies this directly.

If a folder is selected for export, only the zip format is supported. In this case, each data element in that folder will be exported to CLC format, and all these files will be compressed in a single zip file.

CLC format files, or zip files containing CLC format data, can be imported directly into a workbench using the Standard Import tool and selecting "Automatic import" in the Options area.

Backing up and sharing data

If you are backing up data, or plan to share data with colleagues who have a CLC Workbench, exporting to CLC format is usually the best choice. All information associated with that data element will then be available when the data is imported again. CLC format is also recommended when sharing data with the QIAGEN Bioinformatics Support team.

If you are planning to share your data with someone who does not have access to a licensed *CLC Main Workbench* but just wishes to view the data, then you may still wish to export to CLC format. A *CLC Main Workbench* can be run without a license in Viewing Mode, and CLC format data can be imported in the same way it would be using a licensed Workbench. Viewing Mode is described further in section [1.4.7](#).

Compatibility of the CLC data format between Workbench versions

When exporting to CLC or zip format, an option called **Maximize compatibility with older CLC products** is presented at the **Specify export parameters** step, as can be seen in figure [6.9](#). With this option checked, data will be exported without internal compression. Data exported with this option turned on may be larger than it would be otherwise.

Internal compression of CLC data was introduced in CLC Genomics Workbench 12.0, CLC Main Workbench 8.1 and CLC Genomics Server 11.0. This feature decreased the size of the data elements created by the software, but CLC format files with internal compression are not compatible with older versions of the software. Thus, this option should be enabled when exporting data intended for use with older CLC software versions. It should not be needed otherwise.

Internal compression is not required for compatibility with Workbench versions released after this feature was introduced. Data generated in older Workbench versions can, in general, be imported into newer Workbench versions. We endeavor to maintain backwards compatibility of CLC format files whenever possible, so that most CLC format files made using an older version of a Workbenches can be imported into newer Workbench versions.

Internal data compression can be turned off, so no data created is internally compressed. How to do this is described in the Workbench Preferences documentation section [4.4](#).

6.2.5 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 **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

and selecting "Automatic import" in the Options area.

Compatibility of the CLC data format between Workbench versions

Internal compression of CLC data was introduced in CLC Genomics Workbench 12.0, CLC Main Workbench 8.1 and CLC Genomics Server 11.0. If you are sharing data for use in software versions older than these, then please use the standard Export functionality, selecting all the data elements, or folders of elements, to export and choosing either CLC or zip format as the export format. Further information about this is provided in section 6.2.4.

6.2.6 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 or to a CSV file. 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** or History CSV as the format to export to (figure 6.10).
- 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 (figure 6.11).
- Select where the data should be exported to.
- Click on the button labeled **Finish**.

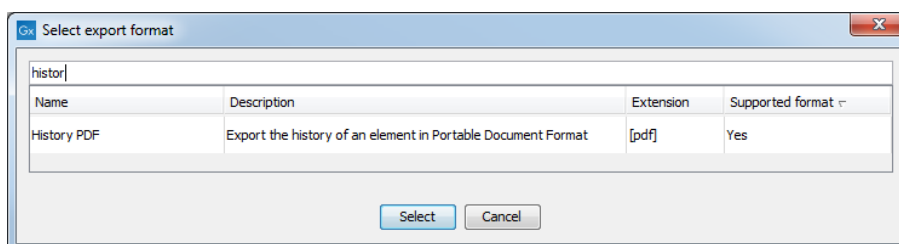


Figure 6.10: Select "History PDF" for exporting the history of an element as a PDF file.

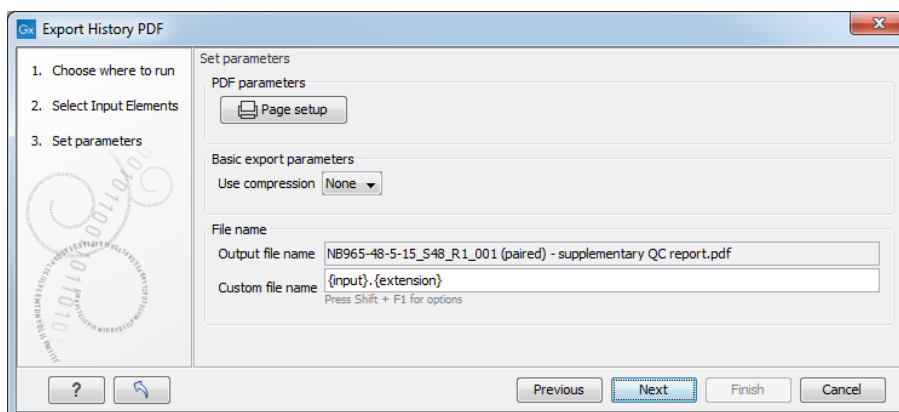



Figure 6.11: When exporting the history in PDF, it is possible to adjust the page setup.

6.2.7 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://resources.qiagenbioinformatics.com/manuals/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.

A note about compatibility Internal compression of CLC data was introduced in CLC Genomics Workbench 12.0, CLC Main Workbench 8.1 and CLC Genomics Server 11.0. If you are backing up data that may be used in software versions older than these, then please select the export option **Maximize compatibility with older CLC products**. Further information about this is provided in section [6.2.4](#)

You can import the zip file into a Workbench by going to:

File | Import (📁) | **Standard Import**

and selecting "Automatic import" in the Options area.

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

6.2.8 Export of tables

Tables can be exported in four different formats; CSV, tab-separated, Excel, or html.

When exporting a table in CSV, tab-separated, or Excel format, numbers with many decimals are printed in the exported file with 10 decimals, or in 1.123E-5 format when the number is close to zero.

Excel limits the number of hyperlinks in a worksheet to 66,530. When exporting a table of more than 66,530 rows, Excel will "repair" the file by removing all hyperlinks. If you want to keep the hyperlinks valid, you will need to export your data to several worksheets in batches smaller than 66,530 rows.

When exporting a table in html format, data are exported with the number of decimals that have been defined in the workbench preference settings. When tables are exported in html format from the server or using command line tools, the default number of exported decimals is 3.

The Excel exporters, the CSV and tab delimited exporters, and the HTML exporter have been extended with the ability to export only a sub-set of columns from the object being exported. Uncheck the option "Export all columns" and click next to see a new dialog window in which columns to be exported can be selected (figure [6.12](#)).

You can choose to "Export the table as currently shown": This will export the table as shown in the active view, including any filtering, sorting, and dynamically added columns.

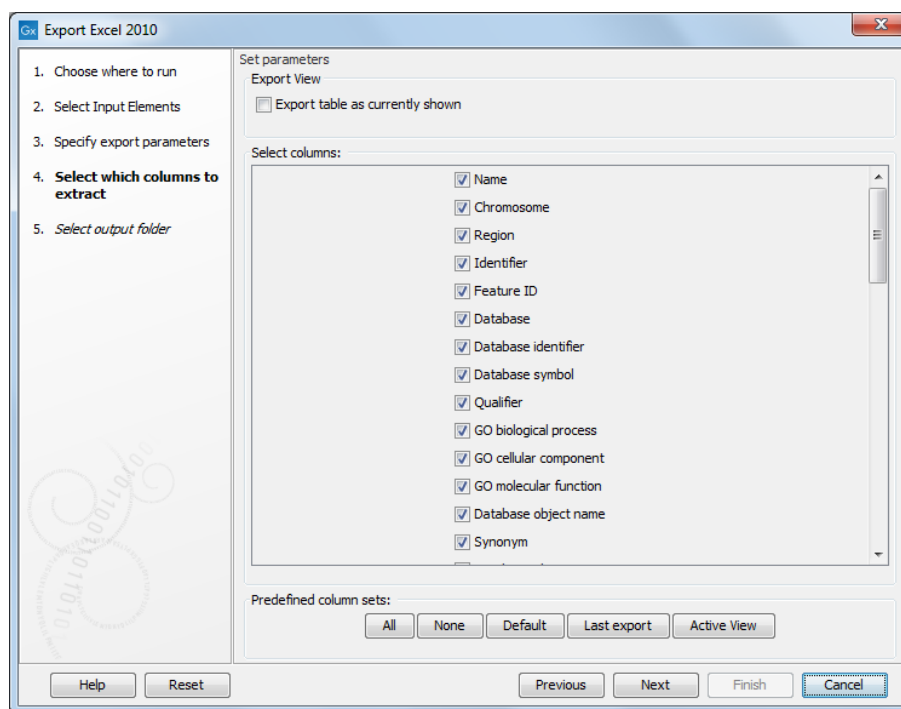


Figure 6.12: Selecting columns to be exported.

You can also choose which columns to export one by one, or choose a predefined subset of columns:

- All: will select all possible columns.
- None: will clear all preselected column.
- Default: will select the columns preselected by default by the software.
- Last export: will select all windows that were selected during the last export.
- Active view (only if a table is currently open): the columns exported are the same than the ones selected in the Side Panel of the table.

After selecting columns, the user will be directed to the output destination wizard page.

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 4.1). 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.

6.3 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 6.13.

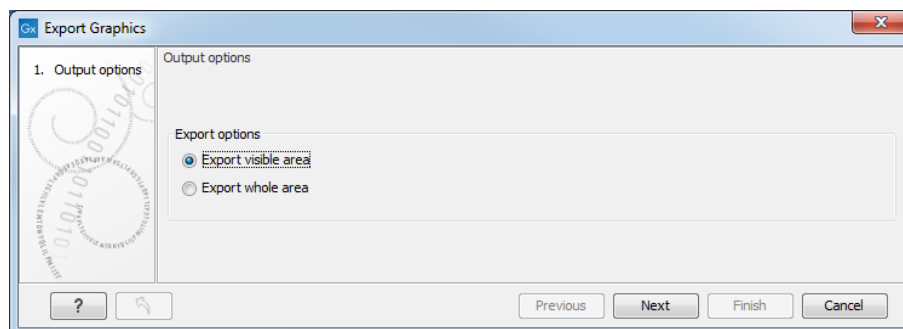


Figure 6.13: Selecting to export whole view or to export only the visible area.

In the following 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 6.14 is a view of a circular sequence which is zoomed in so that you can only see a part of it.

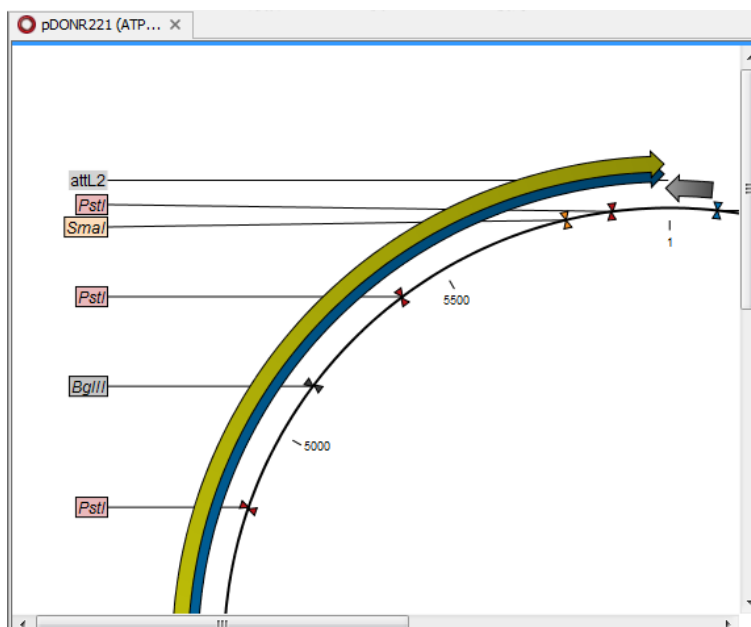


Figure 6.14: A circular sequence as it looks on the screen when zoomed in.

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 6.14 and choosing **Export visible area** can be seen in figure 6.15.

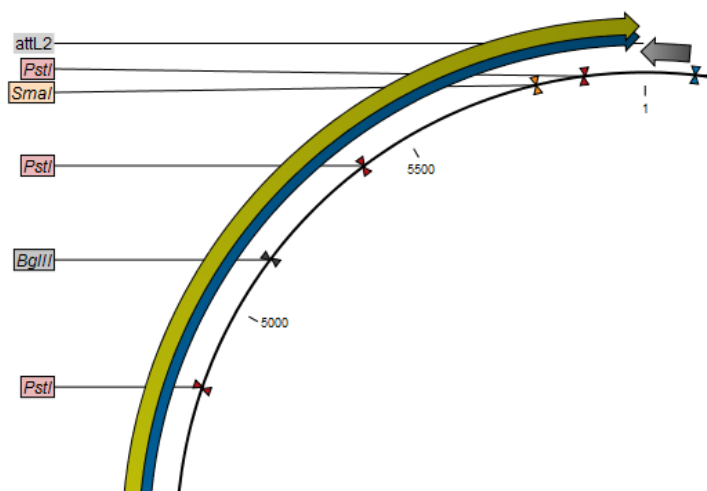


Figure 6.15: The exported graphics file when selecting *Export visible area*.

On the other hand, if you select **Export whole view**, you will get a result that looks like figure 6.16. This means that the graphics file will also include the part of the sequence which is not visible when you have zoomed in.

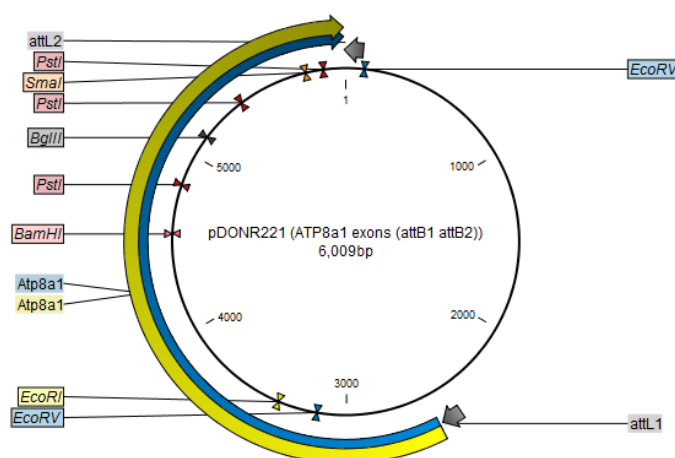


Figure 6.16: The exported graphics file when selecting *Export whole view*. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.

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**).

Finally, choose a name and save location for the graphics file. Then you can either click **Next** or **Finish**, depending on what is available: 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).

6.3.1 File formats

CLC Main Workbench supports the following file formats for graphics export:

Format	Suffix	Type
Portable Network Graphics	.png	bitmap
JPEG	.jpg	bitmap
Tagged Image File	.tif	bitmap
PostScript	.ps	vector graphics
Encapsulated PostScript	.eps	vector graphics
Portable Document Format	.pdf	vector graphics
Scalable Vector Graphics	.svg	vector graphics

These formats can be divided into bitmap and vector graphics. The difference between these two categories is described below:

Bitmap images In a bitmap image, each dot in the image has a specified color. This implies, that if you zoom in on the image there will not be enough dots, and if you zoom out there will be too many. In these cases the image viewer has to interpolate the colors to fit what is actually looked at. A bitmap image needs to have a high resolution if you want to zoom in. This format is a good choice for storing images without large shapes (e.g. dot plots). It is also appropriate if you don't have the need for resizing and editing the image after export.

Parameters for bitmap formats For bitmap files, clicking **Next** will display the dialog shown in figure 6.17.

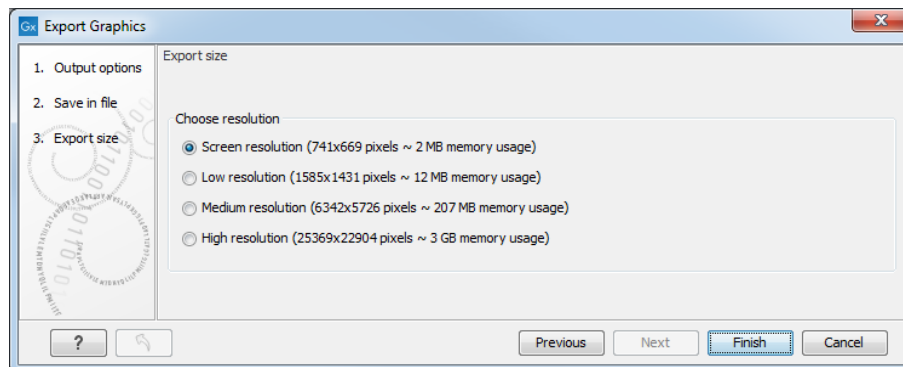


Figure 6.17: Parameters for bitmap formats: size of the graphics file.

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.

Vector graphics Vector graphic is a collection of shapes. Thus what is stored is 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 graphs and reports, but less usable for 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 such as 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 6.1.1 for more about importing external files into *CLC Main Workbench*.

Parameters for vector formats For PDF format, the dialog shown in figure 6.18 will sometimes appear after you have clicked finished (for example when the graphics use more than one page, or there is more than one PDF to export).

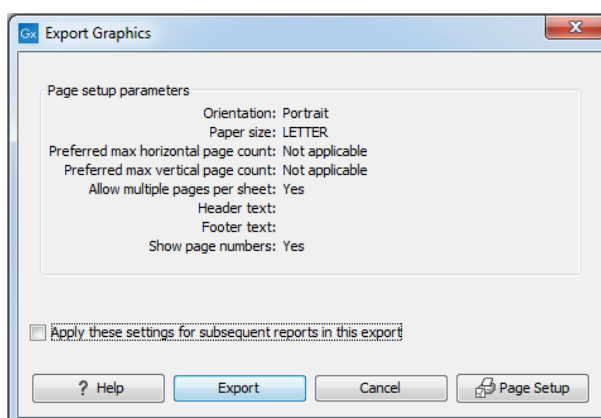


Figure 6.18: Page setup parameters for vector formats.

The settings for the page setup are shown. Clicking the **Page Setup** button will display a dialog where these settings can be adjusted. This dialog is described in section 5.2.

It is then possible to click the option "Apply these settings for subsequent reports in this export" to apply the chosen settings to all the PDFs included in the export for example.

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.

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.

6.4 Export graph data points to a file

Data points for graphs displayed along the sequence or along an alignment or mapping can be exported to a semicolon-separated text file (csv format). An example of such a graph is shown in figure 6.19, showing the conservation score of reads in a read mapping.

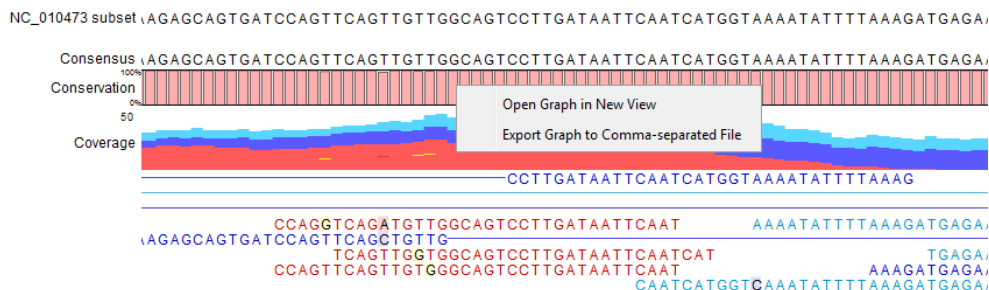


Figure 6.19: A conservation graph displayed along mapped reads. Right-click the graph to export the data points to a file.

To export the data points for the graph, right-click the graph and choose **Export Graph to Comma-separated File**. Depending on what kind of graph you have selected, different options will be shown: If the graph is covering a set of aligned sequences with a main sequence, such as read mappings and BLAST results, the dialog shown in figure 6.20 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.

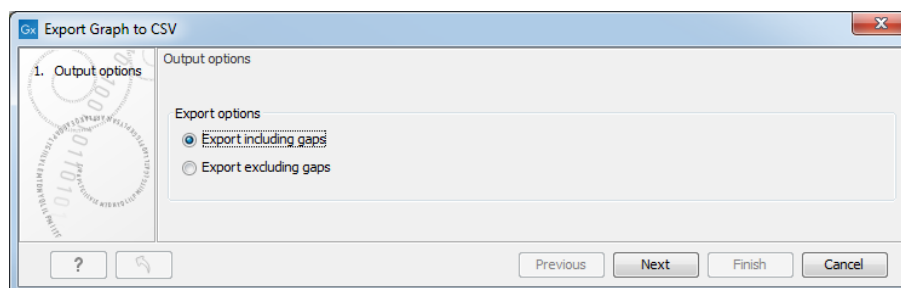


Figure 6.20: Choosing to include data points with gaps

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";
...
```

6.5 CLC Server data import and export

Data export from a CLC Server

When the Workbench is connected to a *CLC Server*, data held on the *CLC Server* can be exported to the system a *CLC Workbench* is running on, or it can be exported to an area the *CLC Server* has been configured to have access to. Such areas are called "Import/export" directories and these must be configured by your server administrator.

To export data to a place the *CLC Workbench* has access to, choose to run the export task on the *Workbench*. To export data to an "Import/Export" directory, choose to run the export task on the *CLC Server*, or *Grid*, as is appropriate for your setup.

Data import to a CLC Server

When connected to a *CLC Server*, data can be imported from "Import/export" directories that have been configured for the *CLC Server*. On some systems, data can also be imported from areas available to your *CLC Workbench*. When that is allowed, you will be able to choose where the files to be imported from are, either "On my local disk or a place I have access to" or "On the server or a place the server has access to", as shown in figure 6.21. The former refers to files available to the *CLC Workbench*, and the latter to files available to the *CLC Server*.

Not all server setups are configured to allow import from local disks, in which case, at least one "Import/export" directory will need to be configured by your server administrator to support import to the *CLC Server*.

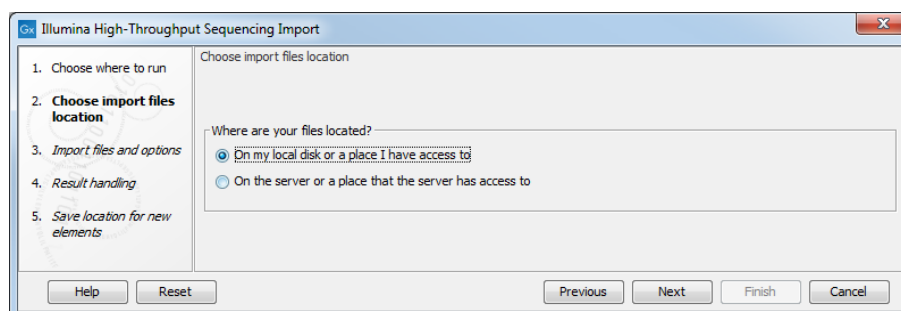


Figure 6.21: Importing data using a Server.

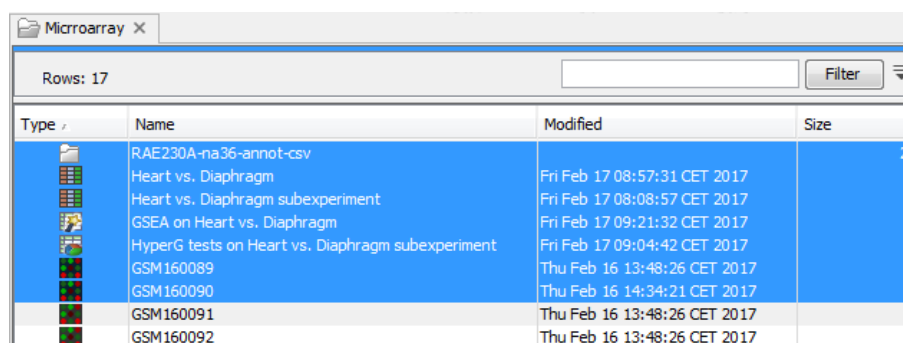
If you choose the option "On my local disk or a place I have access to" when launching an import task, then the *Workbench* must maintain its connection to the *CLC Server* during the first part of the import process, data upload. Further details about this can be found in section 2.3.

6.6 Copy/paste view output

The content of tables (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 in various 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.

Right click a folder in the *Navigation Area* and chooses **Show | Content**. The different elements saved in that folder are now listed in a table in the *View Area*. Select one or more of these elements and use the **Ctrl + C** (or **⌘ + C**) command to copy the selected items.

See figure 6.22.



The screenshot shows a window titled 'Microarray' with a table of 17 rows. The table has four columns: 'Type', 'Name', 'Modified', and 'Size'. The first row is selected and highlighted in blue. The 'Size' column for the selected row shows the number '2'. The 'Name' column contains various file names related to microarray data analysis, including 'RAE230A-ma36-annot-csv', 'Heart vs. Diaphragm', 'Heart vs. Diaphragm subexperiment', 'GSEA on Heart vs. Diaphragm', 'HyperG tests on Heart vs. Diaphragm subexperiment', and several 'GSM' files.

Type	Name	Modified	Size
	RAE230A-ma36-annot-csv		2
	Heart vs. Diaphragm	Fri Feb 17 08:57:31 CET 2017	
	Heart vs. Diaphragm subexperiment	Fri Feb 17 08:08:57 CET 2017	
	GSEA on Heart vs. Diaphragm	Fri Feb 17 09:21:32 CET 2017	
	HyperG tests on Heart vs. Diaphragm subexperiment	Fri Feb 17 09:04:42 CET 2017	
	GSM160089	Thu Feb 16 13:48:26 CET 2017	
	GSM160090	Thu Feb 16 14:34:21 CET 2017	
	GSM160091	Thu Feb 16 13:48:26 CET 2017	
	GSM160092	Thu Feb 16 13:48:26 CET 2017	

Figure 6.22: Selected elements in a Folder Content view.

Then, in a new Excel document, right-click in the cell A1 and paste the items previously copied.

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 7

Data download

Contents

7.1 Search for Sequences at NCBI	132
7.1.1 NCBI search options	133
7.1.2 Handling of NCBI search results	134
7.2 Search for PDB Structures at NCBI	134
7.2.1 Structure search options	135
7.2.2 Handling of NCBI structure search results	136
7.2.3 Save structure search parameters	137
7.3 Search for Sequences in UniProt (Swiss-Prot/TrEMBL)	138
7.3.1 UniProt search options	138
7.3.2 Handling of UniProt search results	139
7.3.3 Save UniProt search parameters	140
7.4 Sequence web info	141


CLC Main Workbench offers different ways of searching and downloading online data. You must be online when initiating and performing the following searches.

7.1 Search for Sequences at NCBI

This section describes searches for sequences in GenBank - the **NCBI** database using Entrez (see <https://www.ncbi.nlm.nih.gov/books/NBK3837/>)

Download | Search for Sequences at NCBI () or **Ctrl + B** ( + **B** on Mac)

This opens the following view (figure 7.1).

The search view can be saved either using dragging the search tab 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.

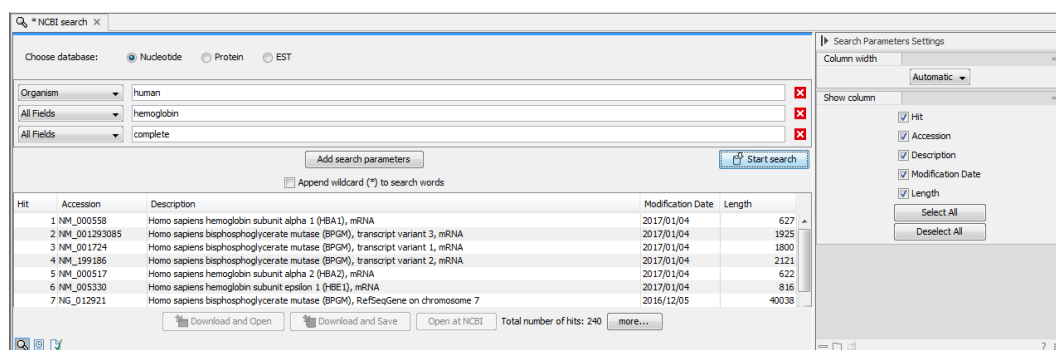


Figure 7.1: The GenBank search view.

7.1.1 NCBI search options

Conducting a search in the **NCBI Database** from *CLC Main Workbench* corresponds to conducting the search on NCBI's website, while having the results available and ready to work with straight away.

You can choose whether you want to search for nucleotide sequences, protein sequences or EST databases.

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. 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.

The following parameters can be added to the search:

- **All fields** Searches in all parameters in the NCBI database at the same time. It also provide an opportunity to search to parameters which are not listed in the dialog (e.g., CD9 NOT homo sapiens).
- **Organism**
- **Definition/Title**
- **Modified Since** Choose one option from the drop-down menu, between 30 days and 10 years.
- **Gene Location** Choose from Genomic DNA/RNA, Mitochondrion, or Chloroplast.
- **Molecule** Choose from Genomic DNA/RNA, mRNA or rRNA.
- **Sequence Length** enter a number for a maximum or minimum length of the sequence.
- **Gene Name**
- **Accession**

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".

A "Feature Key" option is available in GenBank when searching for nucleotide sequences: writing `gene[Feature key] AND mouse` will generate hits for one or more genes and where 'mouse'

appears somewhere in GenBank file. 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**. 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.

7.1.2 Handling of NCBI 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 4). 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 4.6.

Several sequences can be selected, and by clicking the buttons in the bottom of the search view - or right clicking on the selected sequence(s) - you can do the following:

- **Download and Open** opens the sequence in a new view.
- **Download and Save** lets you choose location for saving sequence.
- **Open at NCBI** opens an internet browser and displays the sequence on NCBI's web page.

Double-clicking a hit will download and open the sequence. The hits can also be downloaded into the **View Area** or the **Navigation Area** from the search results by drag and drop or copy/paste.


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.

7.2 Search for PDB 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 12.2.

The NCBI search view is opened in this way:

Download | Search for PBD Structures at NCBI

or **Ctrl + B** ( + **B** on Mac)

This opens the view shown in figure 7.2:

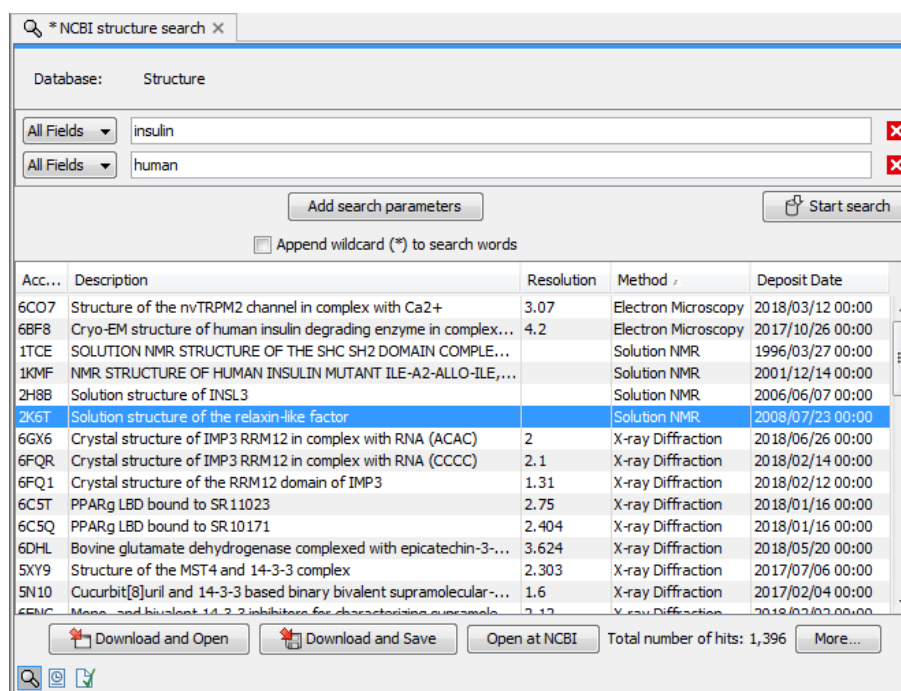


Figure 7.2: The structure search view.

7.2.1 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 GenBank when searching for nucleotide structures. For more information about how to use this syntax, see http://www.ncbi.nlm.nih.gov/entrez/query/static/help/Summary_Matrices.html#Search_Fields_and_Qualifiers

When you are satisfied with the parameters you have entered click **Start search**.

Note! When conducting a search, no files are downloaded. Instead, the program produces a list of links to the files in the NCBI database. This ensures a much faster search.

7.2.2 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 4)). 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 4.6.

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 7.3). 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.

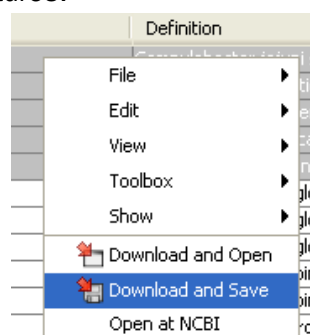


Figure 7.3: By right-clicking a search result, it is possible to choose how to handle the relevant structure.

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.

7.2.3 Save structure search parameters

The search view can be saved either using dragging the search tab 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.

7.3 Search for Sequences in UniProt (Swiss-Prot/TrEMBL)

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 7.4) is opened in this way:

Download | Search for Sequences in UniProt (🔍)

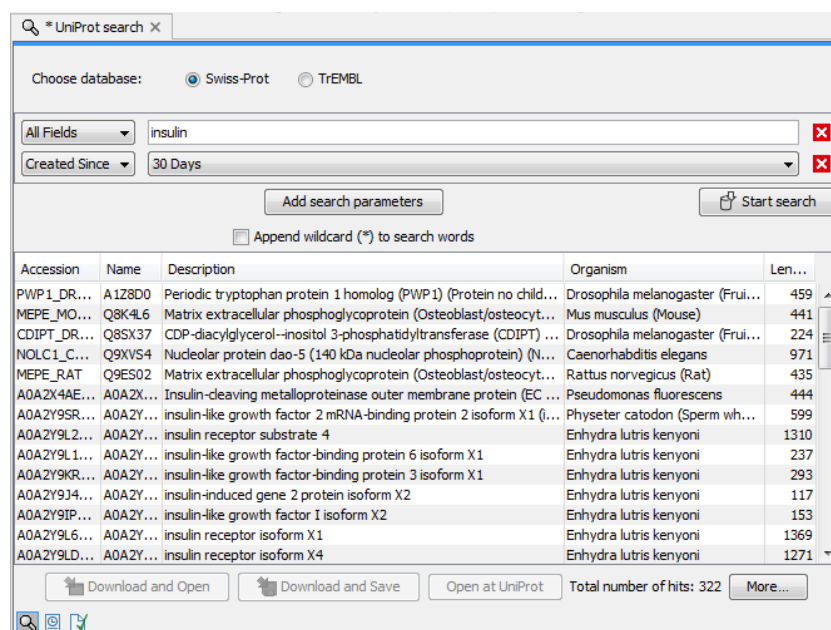


Figure 7.4: The UniProt search view.

7.3.1 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 curated 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.

7.3.2 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 4). 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 4.6.

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. 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.

7.3.3 Save UniProt search parameters

The search view can be saved either using dragging the search tab 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.

7.4 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 11.5). 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 7.5):

Open a sequence or a sequence list | Right-click the name of the sequence | Web Info (🌐) | select the desired search function

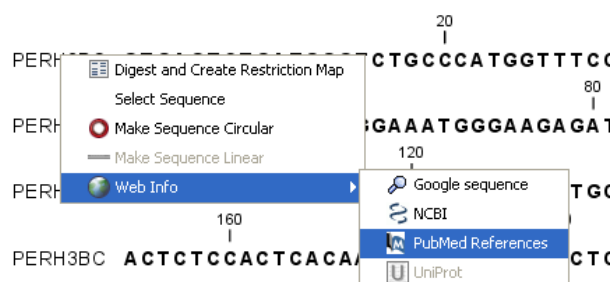


Figure 7.5: Open webpages with information about this sequence.

This will open your computer's default browser searching for the sequence that you selected.

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>.

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.

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 see a dialog and the browser will not open.

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.

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 8

Running tools, handling results and batching

Contents


8.1	Running tools	143
8.2	Handling results	145
8.2.1	Running a tool on a CLC Server	146
8.3	Batch processing	147
8.3.1	Standard batch processing	147
8.3.2	Batch overview	148
8.3.3	Parameters for batch runs	149
8.3.4	Running the analysis and organizing the results	149

This section describes how to run a tool using singles files as input, as well as how to handle and inspect results. We also review how to run tools using the batch mode when the option is enabled.

8.1 Running tools

All the analyses in the **Toolbox** are performed in a step-by-step procedure:

- Data elements to be used in the analysis are selected.
- Any configurations necessary for the tool to run are made.
- The results are opened or saved.

You can open a tool from the Toolbox by double clicking on its name in the Toolbox in the bottom left side of the Workbench, or by selecting it from the Toolbox menu at the top. You can also find tools quickly by clicking on the Launch button () in the toolbar. Double click on the name of the tool in the table to launch it. If you enter a search term in the field at the top of the Quick launch window, tools with that term in their names or descriptions will be listed.

When you open a tool, a wizard pops up in the center of the View Area. Stepping through a succession of wizard steps, you will select the data to analyze, configure any analysis parameters, and specify how the results should be handled. You can navigate between wizard steps by clicking the buttons **Next** and **Previous** at the bottom of the window.

If you have logged into a *CLC Server* from your Workbench, you will first be asked to select whether the job should be run on the Workbench or submitted to the server. These choices, along with information about data selection and other considerations when launching tasks on a *CLC Server* are provided in section 8.2.1.

Generally, the first analysis configuration step involves selecting the data elements to be used as input. A view of your Navigation Area will be presented to you. That view will show data elements appropriate for use as input for that tool. Folders are also shown. For example, in figure 8.1 you can see a the Workbench Navigation Area (on the left) and a view of the same Navigation Area in the wizard (on top) for the Assemble Sequences tool. This tool only accepts nucleotide sequences and nucleotide sequence lists, so data elements of other types that can be seen in the Workbench Navigation Area, such as the one called "Read mapping", and the amino acid sequence ATP8a1, are not displayed in the wizard Navigation Area.

The data types that can be used as input for a given tool are described in the manual section about that tool. This documentation can be opened directly by clicking on the **Help** button in the bottom left corner of the launch wizard.

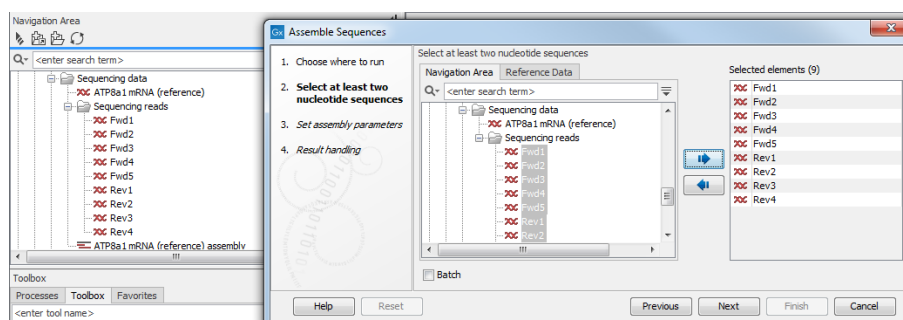


Figure 8.1: You can select input files for the tool from the Navigation Area view presented on the left hand side of the wizard window.

To indicate the data elements to be used in the analysis, either double click on them in the "Navigation Area" view on the left, or select them with a single click and then click on the right hand arrow. These items will then be listed in the "Selected elements" list on the right. If data elements of appropriate types were already selected in the Workbench Navigation Area before launching the tool, these will be automatically entered into the Selected elements list. To remove entries in that list, just double click on them or select them with a single click and then click on the left hand arrow.

When multiple elements are selected, most analysis tools will treat all those elements as a single input data set unless the "Batch" option at the bottom, has been selected. If that option is selected, then the tool will be run multiple times, once for each "batch unit", which may be a data element, or folder containing data elements or containing folders of elements. Batch processing is described in more detail in section 8.3.

Once the data of interest has been selected, click on **Next**. Depending on the tool, there may now be one or more steps for configuring analysis parameters. An example is shown in figure 8.2. Clicking on the **Reset** button resets all parameters in that step to their default values.

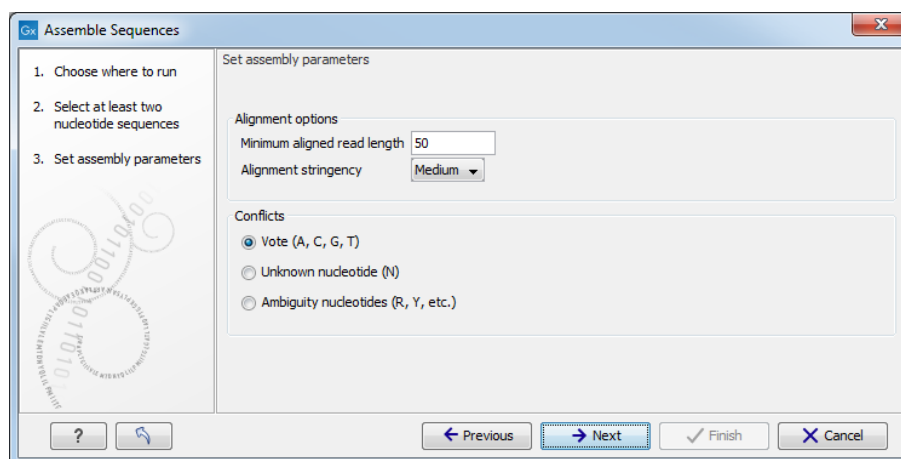


Figure 8.2: An example of a "Set parameters" window.

8.2 Handling results

Some tools can generate several outputs. If there is a choice of which ones to generate, you will be able to configure this in the final wizard step, called "Result handling". The kind of output files generated by a tool are described in the tool specific sections of the manual.

For tasks run on a Workbench (as opposed to a CLC Server) the "Result handling" window also allows you to decide whether you want to open or save your results.

- **Open.** This will open the result of the analysis in a view. This is the default setting.
- **Save** The results will be saved rather than opened. You will be prompted for where you wish the results to be saved (figure 8.3). You can save to an existing area or create a new folder to save the results into.

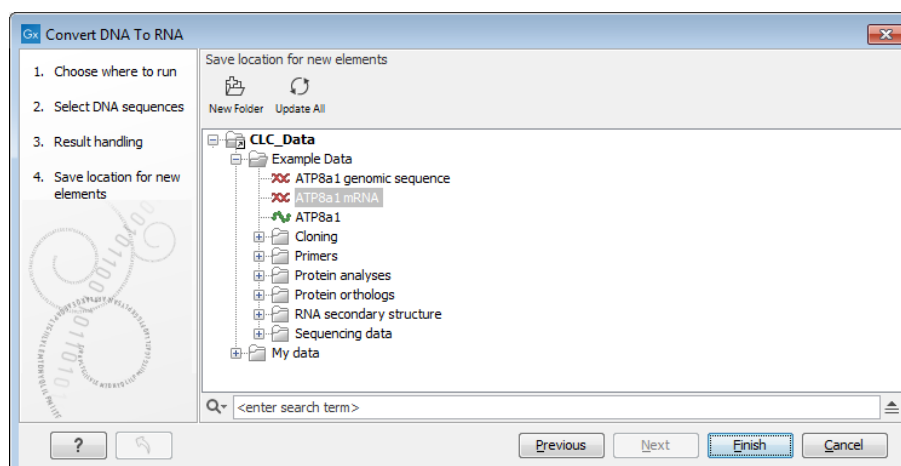


Figure 8.3: Specify where to save the results of an analysis.

You may also have an option called "Open log". If checked, a window will open in the View area after the analysis has started and the progress of the job will be reported there line by line.

Click on the button labeled **Finish** to start the analysis.

If you chose the option to open the results, they will open automatically in one or several tabs in the View Area. The data will not have been saved at this point. The name of each tab is in bold,

appended with an asterisk to indicate this. There are several ways to save the results you wish to keep:

- Drag the tab to the relevant location in the Navigation Area.
- Select the tab and then use the key combination Ctrl + S (or ⌘ + S on macOS).
- Right click on the tab and choose "Save" from the context menu.
- Use the "Save" button in the Workbench toolbar.
- Go to the File menu and select the option "Save" or "Save As..."

If you chose to save the results, they will have been saved in the location specified. You can open the results in the Navigation Area directly after the analysis is finished. A quick way to find the results is to click on the little arrow to the right of the analysis name in the Processes tab and choose the option "Show results" or "Find Results", as shown in figure 8.4.

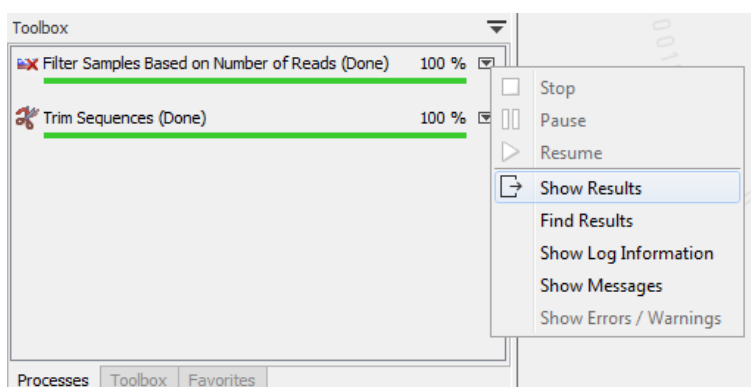


Figure 8.4: Find or open the analysis results by clicking on the little arrow to the right of the analysis name in the Processes tab and choosing the relevant item from the menu.

8.2.1 Running a tool on a CLC Server

When you launch an analysis from a Workbench that is logged into a CLC Server, you are offered the choice of where the analysis should be run, as shown in figure 8.5.

- **Workbench.** Run the analysis on the computer the CLC Workbench is running on.
- **Server.** Run the analysis using the CLC Server. For job node setups, analyses will be run on the job nodes.
- **Grid.** Only offered if the CLC Server setup has grid nodes. Here, jobs are sent from the master CLC Server to be run on grid nodes. The grid queue to submit to can be selected from the drop down list under the Grid option.

You can check the **Remember setting and skip this step** option if you wish to always use the selected option when submitting analyses that can be run on a CLC Server. If you select this option but later change your mind, just start up an analysis and click on the **Previous** button. The step with the options for where to run analyses will then be shown.

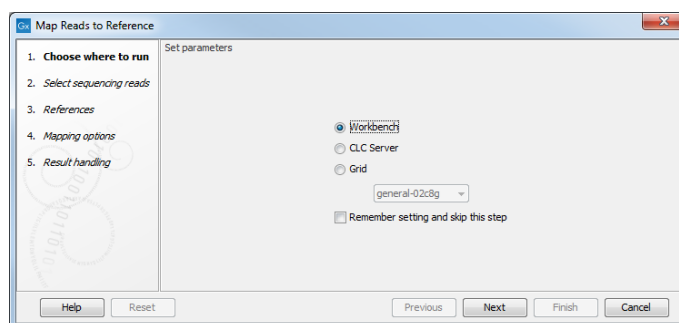


Figure 8.5: When logged into the CLC Server, you can select where a job should be run.

When launching a task to run on a CLC Server or on grid nodes, there are a few things to be aware of:

- You can only analyze data stored in CLC Server data areas. Thus only these data areas will be offered to select from when configuring an analysis.
- You have to save the analysis results. For a single analysis run on the Workbench, you can normally choose how to handle the results: **Open** or **Save**. Results from analyses performed on a CLC Server must be saved.
- After you have launched an analysis, it is submitted to the CLC Server to be handled. You can then close the Workbench or disconnect from the CLC Server if you wish. If an analysis finishes while your Workbench is closed or not connected to the CLC Server, you will see a notification about this when you next log in from the Workbench.
- When importing data into a CLC Server, the location of the data being imported affects when it is safe to close the Workbench or disconnect from the server. Further information about that can be found in section 2.3.

8.3 Batch processing

Batch processing refers to running an analysis multiple times, using different inputs for each analysis run. For example, if you have 10 sequence lists and wish to run 10 mapping analyses, one per sequence list, then these 10 analyses could be launched by setting up one batch job. When a job is run in batch mode, parameter settings stay the same for each run. It is just the inputs that are changed.

This section describes batch processing as it applies to most workbench tools and to workflows with a single input element.

Batching installed workflows with multiple input elements, where **all** input elements will be changed per batch, is done differently (see section 9.5).

8.3.1 Standard batch processing

Standard batch mode is activated by clicking the **Batch** checkbox in the dialog where the input data is selected (figure 8.6).

Unlike launching a single task, you can select a folder as well as, or instead of, individual data elements for the analysis.

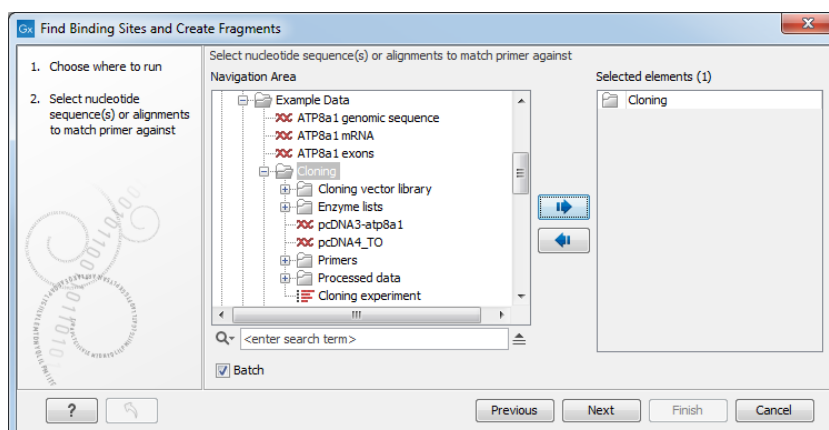


Figure 8.6: The Cloning folder includes both folders and sequences.

A batch unit is the set of data that will be used as a single input set for a given run of an analysis. A given batch unit can consist of one or more data elements.

If a folder is selected as input to a batch analysis, each folder or data element directly under that folder will be considered a batch unit. This means:

- Each individual data element contained directly within the folder is a batch unit.
- Each subfolder directly within this folder is a batch unit, so all elements within a given subfolder will be considered as single input for the purposes of the analysis.
- Elements in any more deeply nested subfolders (e.g. subfolders of subfolders of the originally selected folder) will not be considered for the analysis.

8.3.2 Batch overview

The next Wizard step is the batch overview where you have the opportunity to refine the list of data that will be in each batch unit. For example, you could use this step to ensure that only trimmed sequence lists - and not all sequence lists - should be used for the analysis that is being setup.

The batch overview lists the batch units on the left and the contents of the selected batch unit on the right (figure 8.7).

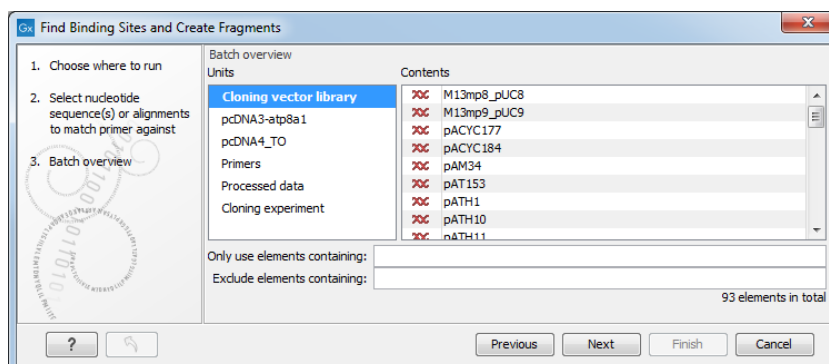


Figure 8.7: Overview of the batch run. At the bottom right, the number of files to be analysed, summed across all batch units, is shown, 92 in this case.

In this example, the two sequences (pcDNA) are defined as separate batch units because they are located at the top level of the Cloning folder. Of the four subfolders of the Cloning folder initially selected, three are listed in this view. In each of these subfolders, any data elements that the analysis could use as input will be used unless action is taken at this point to exclude some of these. So all the elements in the subfolder "Cloning vector library" and shown on the right-hand side of the dialog will be included as part of a single analysis run .

Note that folders that do not contain any data that can be used by the tool being launched will not be shown in that dialog.

Including and excluding data elements in batch units There are three ways to refine the data elements that should be included in a batch unit, and thereby get taken forward into the analysis.

- **Use the fields labeled Only use elements containing and Exclude elements containing at the bottom of the batch overview** This refinement is done based on data element names. for example, only paired reads might be desired for the analysis, in which case, putting the text "paired" into the **Only use elements containing** field might be useful.
- **Remove a whole batch unit** Right-click on the batch unit to be removed and choose the option **Remove Batch Unit**.
- **Remove a particular data element from a batch unit** Right click on the element of a batch unit to be removed and choose the option **Remove Element**. This can be useful when filtering based on name, described in the first option, cannot be used to refine the batch units specifically enough.

8.3.3 Parameters for batch runs

The subsequent dialogs depend on the analysis being run and the data being input. Generally, one of the batch units will be specified as the parameter prototype and will then be used to guide the choices in the dialogs. By default, the first batch unit (marked in bold) is used for this purpose. This can be changed by right-clicking another batch unit and choosing the option **Set as Parameter Prototype**.

When launching tools normally (non-batch runs), the Workbench does much validation of inputs and parameters. When running in batch, this validation is not performed. This means that some analyses will fail if combinations of input data and parameters are not right. Therefore we recommend that batching is used when the batch units are quite homogenous in terms of the type and size of data.

8.3.4 Running the analysis and organizing the results

The last step in setting up a batch analysis is to choose where to save the outputs (figure 8.8).

The options available are:

- **Save in input folder** Save all outputs into the same folder as the input data. If the batch units consisted of folders, then the results of each analysis would be saved into the folder with the data it was generated using. If the batch units were individual data elements, then all the results will be placed into the same folder as those input data elements.

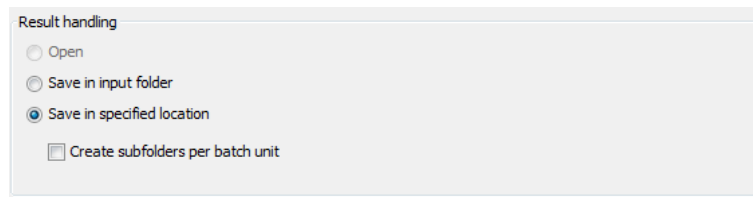


Figure 8.8: Options for saving results when the tool was run in batch.

- **Save in specified location** Choose the folder where the outputs should be saved to, where when:
 - **Create subfolders per batch unit** is **unchecked**, all results for all batch units will be written to the specified folder.
 - **Create subfolders per batch unit** is **checked**, results for each batch unit will be written to a newly created subfolder of the selected folder. One subfolder is created per batch unit.

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 this is different for Workbenches and Servers:

- On a 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 avoids many parallel analyses that would draw on the same compute resources and slow down the computer.
- On a CLC Server, all the processes are placed in the queue, and the queue takes care of distributing the jobs. This means that if the server set-up includes multiple nodes, different batch unit analyses may be run in parallel.

To stop the whole batch run, stop the "master" process. From the Workbench, this can be done by finding the master process in the Processes tab in the bottom left hand corner. Click on the little triangle on the right hand side of the master process and choose the option **Stop**.

For some analyses, there is an extra option in the final step to create a log of the batch process. This log will be created in the beginning of the process and continually updated with information about the results. 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.

Chapter 9

Workflows

Contents

9.1	Creating a workflow	152
9.1.1	Adding workflow elements	152
9.1.2	Configuring workflow inputs	154
9.1.3	Configuring workflow tools	155
9.1.4	Locking and unlocking parameters	158
9.1.5	Connecting workflow elements	158
9.1.6	Output	160
9.1.7	Input	162
9.1.8	Layout	164
9.1.9	Input modifying tools	166
9.1.10	Workflow validation	167
9.1.11	Workflow creation helper tools	169
9.1.12	Adding to workflows	169
9.1.13	Snippets in workflows	170
9.1.14	Change the order of tracks in the Track List	172
9.2	Distributing and installing workflows	173
9.2.1	Creating a workflow installation file	174
9.2.2	Installing a workflow	177
9.2.3	Managing workflows	178
9.2.4	Workflow version and update	180
9.3	Executing a workflow	182
9.4	Open copy of installed workflow	183
9.5	Batch launching workflows with multiple inputs	183

The *CLC Main Workbench* provides a framework for creating, distributing, installing and running workflows. A workflow consists of a series of connected tools where the output of one tool is used as input for another tool. Once the workflow is set up, it can be installed (either in your own Workbench or it can be shared with colleagues and installed on a server). 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.

Note that the examples below are using tools from the *CLC Genomics Workbench* that are not necessarily 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*.

9.1 Creating a workflow

A workflow can be created by pressing the "Workflows" button (🏠) in the toolbar and then selecting "New Workflow..." (🏠).

Alternatively, a workflow can be created via the menu bar:

File | New | Workflow (🏠)

This will open a new view with a blank screen where a new workflow can be created.

9.1.1 Adding workflow elements

First, click the **Add Element** (+) 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 9.1).

Alternatively elements can be dragged directly from the **Toolbox** into the workflow. Only workflow enabled elements can be dropped in the workflow.

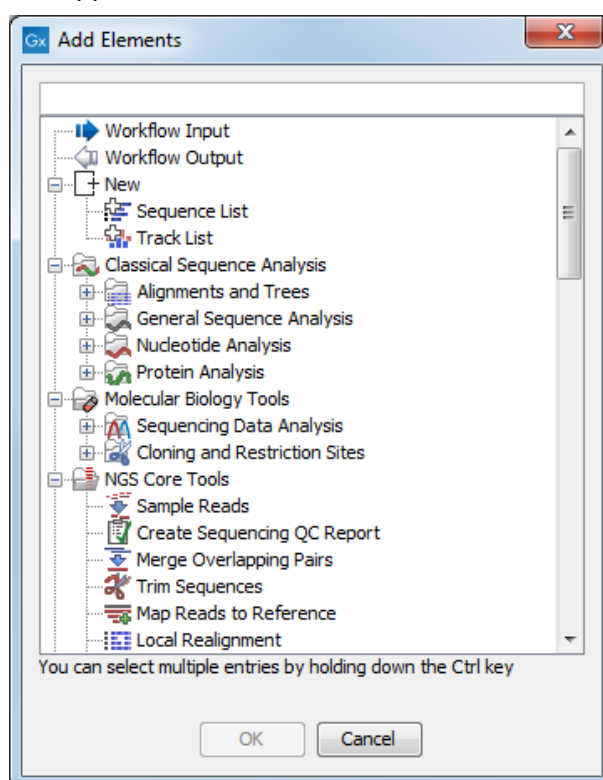


Figure 9.1: Adding elements in the workflow.

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 are explained in section 9.1.7.

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 add more later on if you wish.

You will now see the selected elements in the editor (see figure 9.2).

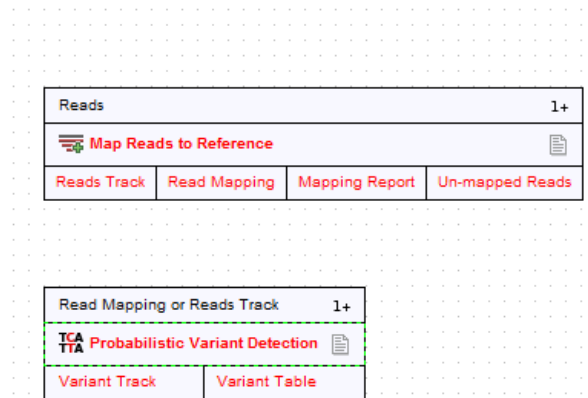


Figure 9.2: Read mapping and variant calling added to the workflow.

Once added, you can move and rearrange the elements by dragging them with the mouse. To do this, click on part of the box containing the name of the element and then, keeping the mouse button depressed, drag the element to the desired position.

9.1.2 Configuring workflow inputs

There are two kinds of inputs in a workflow (figure 9.3):

- Primary inputs: these inputs vary every time the workflow is run. Primary inputs are shown with a light grey background in the workflow editor. An example of a primary input is the case variant track in the example below.
- Secondary inputs: these inputs are considered more constant. Secondary inputs could be reference data, or a set of controls to evaluate the case sample(s) against. Secondary inputs have a brown background in the workflow editor.

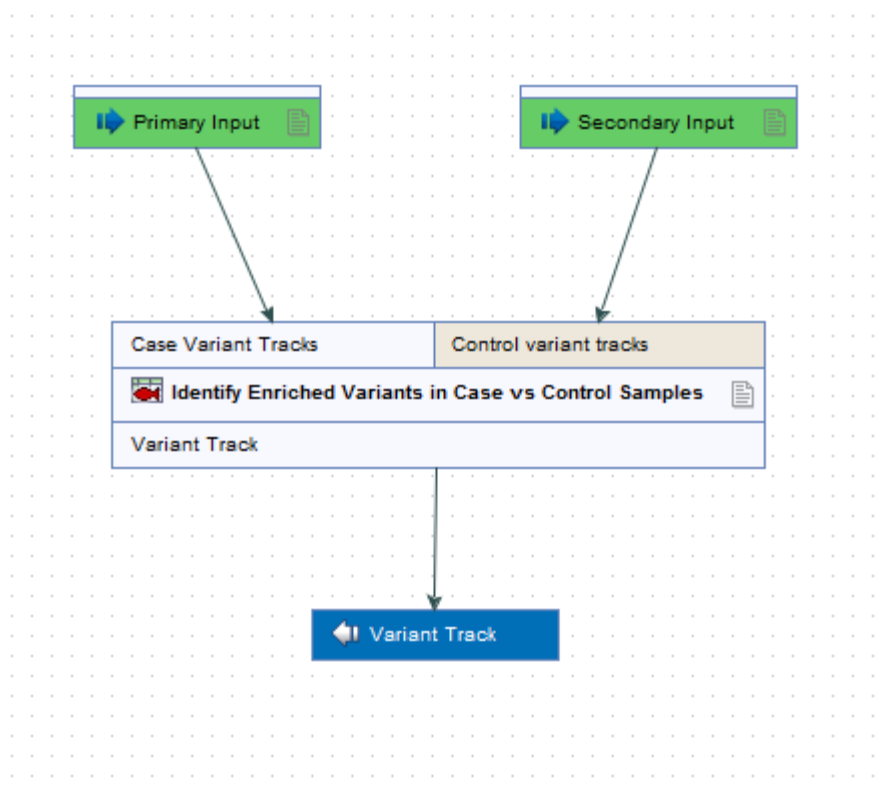


Figure 9.3: *Selecting a particular file to configure the workflow with.*

Secondary inputs can be configured to work in two different situations:

- When running the workflow without a Reference Data Set, or when running it with a reference data set that does not have a reference data element with the corresponding role.
- When running the workflow with a Reference Data Set. In this case, the Workflow Role of the workflow input must match the Workflow Role of a reference data element, in order for that element to be used .

To configure a secondary input, double-click the name of the input. You can now select an element under Workflow Input, which will configure the input for situation 1) (as seen in figure 9.4). Or you can enter a Workflow Role to configure the input for situation 2) (figure 9.5). You can either type in a role name of your choice, or choose one from the drop down list, as long as there will be a reference data set with a matching role.

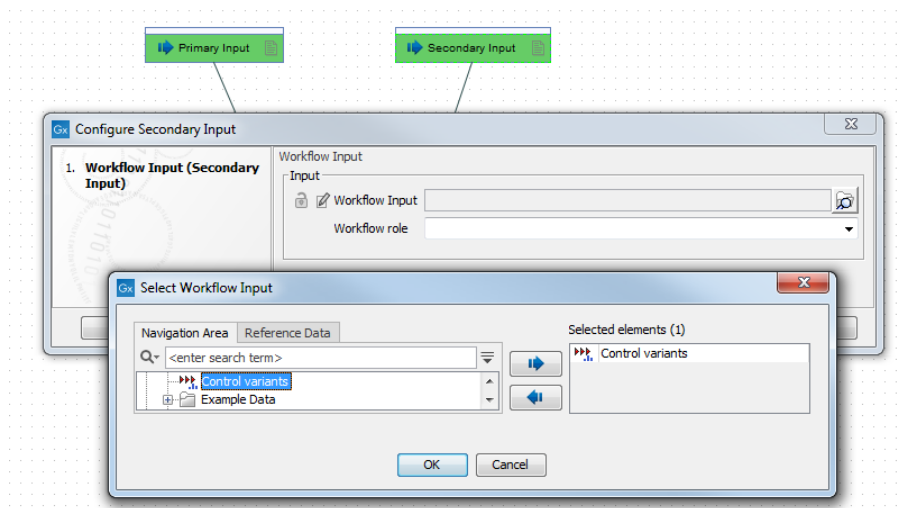


Figure 9.4: Selecting a particular element to configure the workflow with. Remember to lock the configuration before saving the workflow if you want this step not to appear in the workflow's wizard.

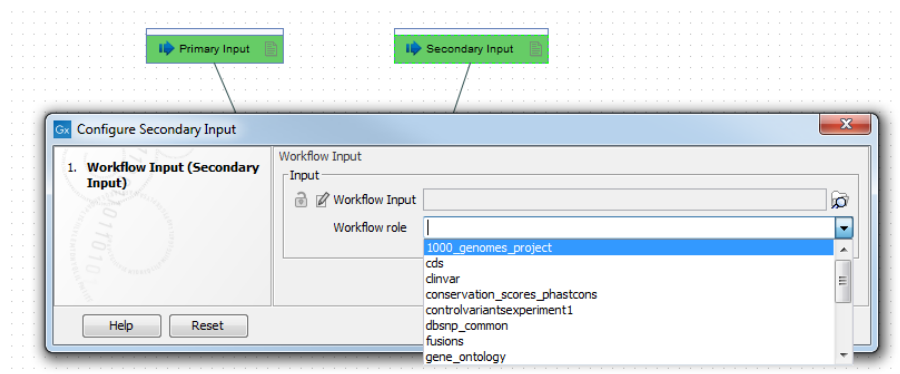


Figure 9.5: Selecting a particular role to configure the workflow with. Remember to lock the configuration before saving the workflow. The workflow's wizard will include a step where you can choose a reference data set.

Note: When running the workflow, the workflow wizard makes it possible to override the configuration of any unlocked Workflow Input. If you want to make sure that the workflow always runs according to your configuration, then lock the configuration by clicking the Lock icon. Learn more about locking parameters in section 9.1.4.

9.1.3 Configuring workflow tools

Each of the tools can be configured by right-clicking the name of the tool as shown in figure 9.6.

The first option you are presented with is the option to **Rename** the element. This can be useful when you wish to discriminate between 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 run. To rename the element, right click on the tool in the workflow and select the "Rename" option, or click on the tool in the workflow and then press the F2 key.

The **Remove** option is used to remove elements from the workflow. The shortcut Alt + Shift + R

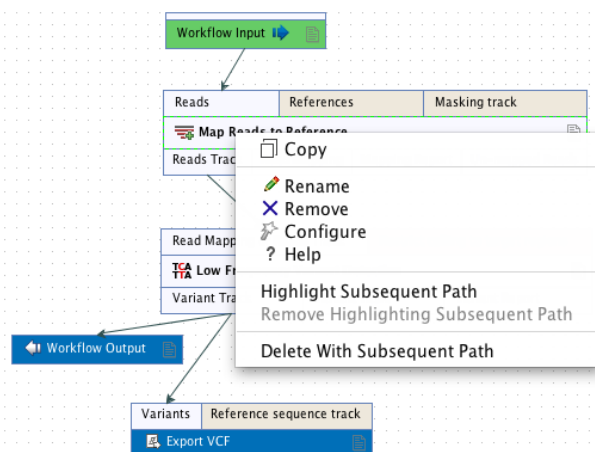


Figure 9.6: Configuring a tool.

removes all elements from the workflow.

You can also configure a given element using the **Configure** option from the right click menu or by double-clicking on the element. This opens a dialog with options for setting parameters, selecting reference data, selecting the export destination of specified columns, etc. An example is shown in figure 9.7.

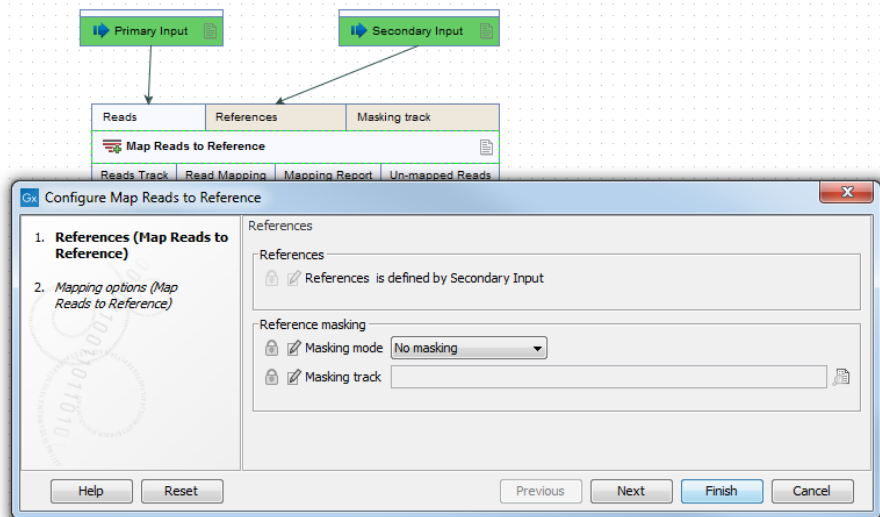


Figure 9.7: Configuring read mapper parameters.

Click through the dialogs using **Next** and press **Finish** when you are done. This saves the parameter settings for that tool. These are then applied when the workflow is executed.

You can also change the name of a parameter if you so wish, for example, to help with usability for the intended users of a workflow. To do this, click on the edit icon (✎) and enter a new name.

The lock icons in the dialog are used for specifying whether the parameter should be locked and unlocked as described in the next section. Locking parameters means that the workflow will be run with the same parameters every time; the user will not be prompted to supply values for locked parameters when they launch the workflow.

Once an element has been configured, the workflow element will be shaded with a darker color to help in distinguishing which elements have been configured.

The **Highlight Subsequent Path** option causes the element that was clicked on, and all the elements downstream of that one, to be highlighted. Other elements will be grayed out (figure 9.8). The **Remove Highlighting Subsequent Path** option reverts the highlighting, returning to the normal workflow layout.

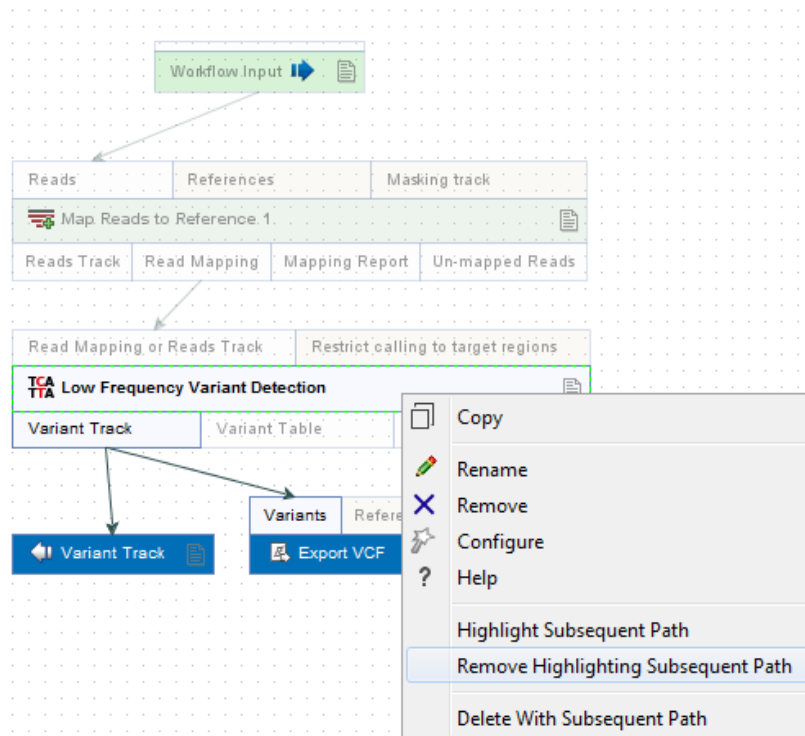



Figure 9.8: Highlight path from the selected tool and downstream.

Similarly, instead of configuring the various tools individually, the **Configuration Editor** enables the specification of all settings, references, masking parameters etc. through a single wizard window (figure 9.9). This editor is accessed through the  icon located in the lower left corner.

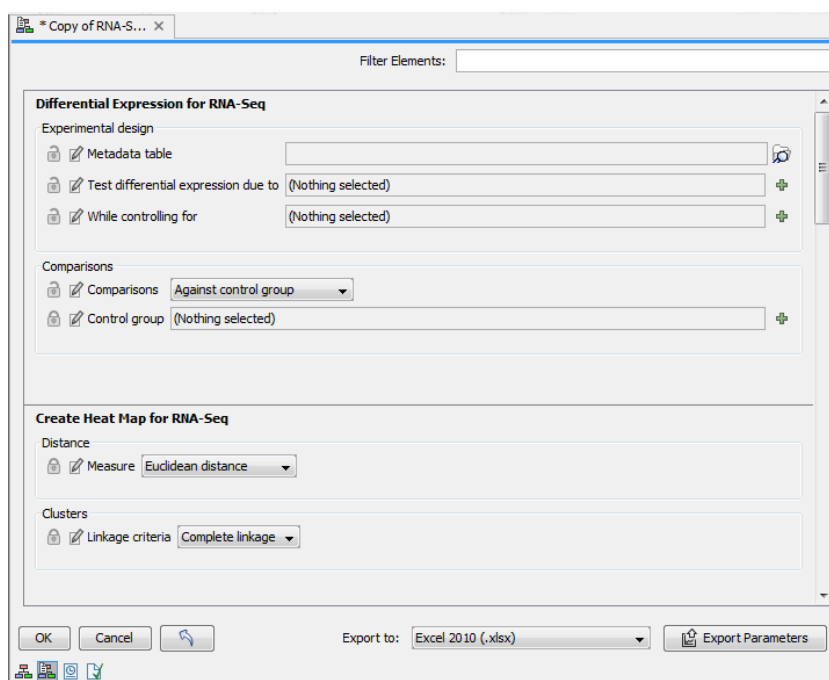


Figure 9.9: The Configuration Editor can be used to configure all the tools that can be configured in a given Workflow.

9.1.4 Locking and unlocking parameters

When creating the workflow, the workflow author can specify which parameters should be locked or unlocked. If a parameter is locked, it cannot be changed in the installation or the execution step. Conversely, if it is left open, that parameter can be changed, either when running the Workflow or when installing it (see section 9.2).

By default, data parameters are unlocked. When installing the workflow on a different system to the one where it was created, the connection to the data needs to be re-established. 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 in the same location as the system where the Workflow was created.

9.1.5 Connecting workflow elements

Figure 9.10 explains the different parts of a workflow element.

Reads	References	Masking track	Input parameters	
Map Reads to Reference			Name of the tool	
Reads Track	Read Mapping	Mapping Report	Un-mapped Reads	Output options

Figure 9.10: A workflow element consists of three parts: input, name of the tool, and output. The primary inputs are colored with a light grey background, and the secondary inputs (parameter inputs) are colored with a brown background.

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 9.2, 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 9.11. 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 9.12).
- 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 9.1, 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.

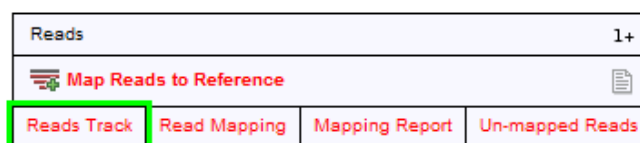


Figure 9.11: Dragging the reads track output with the mouse.

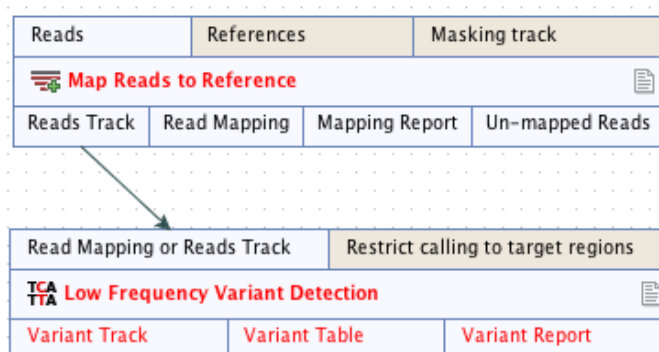


Figure 9.12: The reads track is now used for variant calling.

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 detection tool 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 detection tool.

Figure 9.13 demonstrates how one tool can receive input from two different sources; 1) a reads track that is the input that hold the data that is to be analyzed (in this case reads that is to be locally realigned), and 2) a parameter that can have different functions depending on the tool that it is connected to (in this case the InDel track is used as a guidance track for the local realignment. In other situations the parameter track could be used for e.g. annotation or could provide a reference sequence).

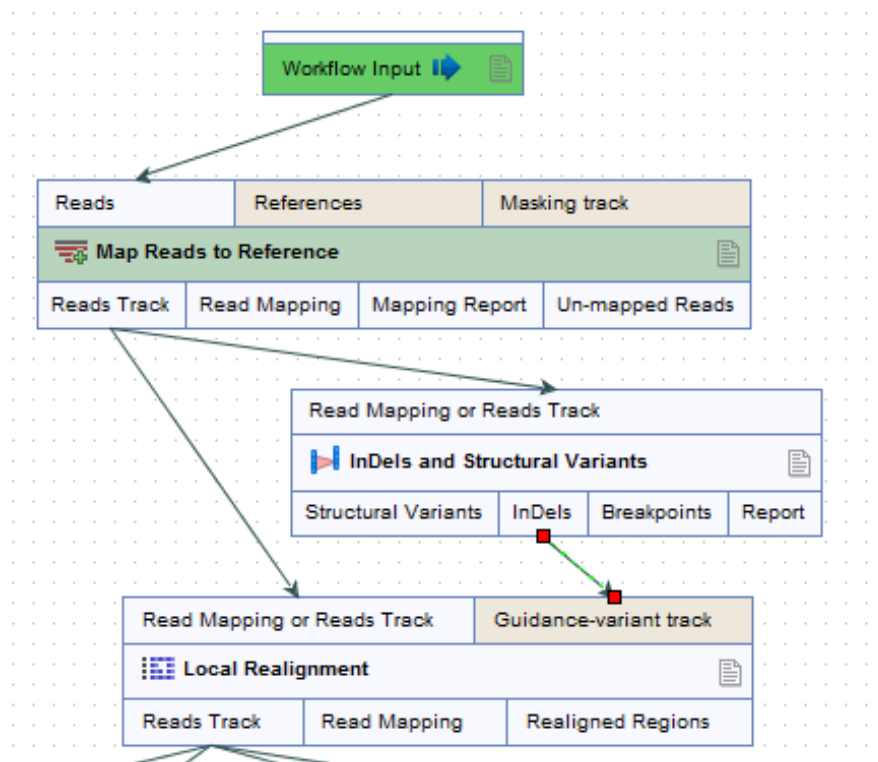


Figure 9.13: A tool can receive input from both the generated output from another tool (in this example a reads track) and from a parameter (in this case indels detected with the InDels and Structural Variants tool).

9.1.6 Output

Besides connecting the elements together, you have to decide what the input and the output of the workflow should be. We will first look at specification of the output, which is done by right-clicking the output box of any tool and selecting **Use as Workflow Output** as shown in figure 9.14.

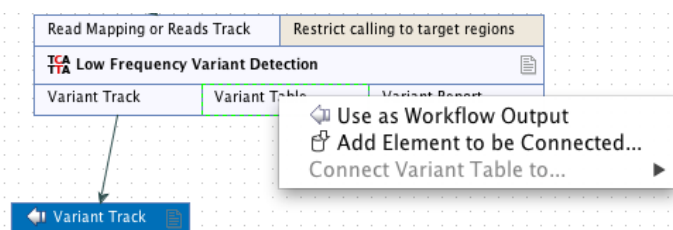


Figure 9.14: Selecting a workflow output.

You can mark several outputs this way throughout the workflow. Note that no intermediate results are saved unless they are marked as workflow output. In fact, when the workflow is executed, all the intermediate results are indeed saved temporarily, but they are automatically deleted when the workflow is completed. However, if a workflow fails, the intermediate results are not deleted and will be found in a folder named after the workflow with the mention "intermediate".

By double-clicking the output box, you can specify how the result should be named as shown in

figure 9.15.

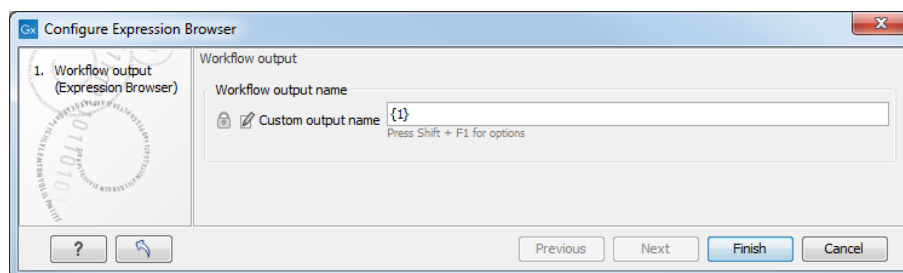


Figure 9.15: Specifying naming of a workflow output.

In this dialog you can enter a name for the output. Dynamic placeholders are available, which can help in setting up specific and standardized names for outputs. If you mouse over the Custom output name field in the dialog, the placeholders are listed. You can also click on Shift+F1 to see the options. The placeholders below are available. They are not case specific.

- **{name}** or **{1}** - default name for the tool's output
- **{input}** or **{2}** - the name of the workflow input (and not the input to a particular tool within a workflow).
- **{user}** - name of the user who launched the job
- **{host}** - name of the machine the job is run on
- **{year}**, **{month}**, **{day}**, **{hour}**, **{minute}**, and **{second}** - timestamp information based on the time an output is created. Using these placeholders, items generated by a workflow at different times can have different filenames.

When deciding on an output name, you can choose any combination of the placeholders, as well as custom names and punctuation, for example, `{input}({day}-{month}-{year})`. A meaningful name to a variant track could be `{2} variant track` as shown in figure 9.16. Here, if your workflow input was named `Sample 1`, the result would be "Sample 1 variant track".

The placeholders available for exports are slightly different than for other workflow outputs and are described in section 6.2.

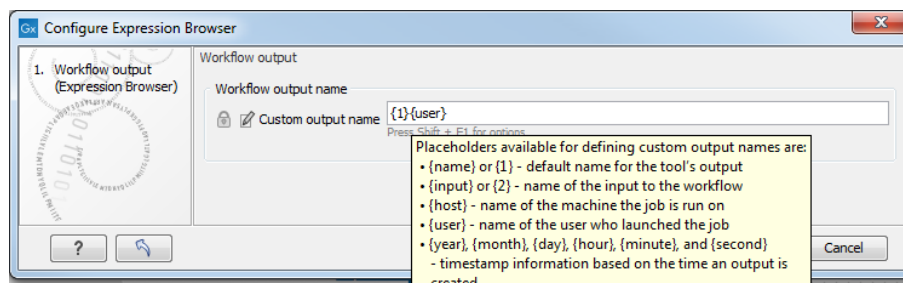


Figure 9.16: Providing a custom name for the result.

It is also possible to save workflow outputs into subfolders by using a forward slash character `/` at the start of the output name definition. For example the custom output name

`/variants/{name}` refers to a folder "variants" that would lie under the location selected for storing the workflow outputs. When defining subfolders for outputs, all later forward slash characters in the configuration, except the last one, will be interpreted as further levels of subfolders. For example, a name like `/variants/level2/level3/myoutput` would put the data item called `myoutput` into a folder called `level3` within a folder called `level2`, which itself is inside a folder called `variants`. The `variants` folder would be placed under the location selected for storing the workflow outputs. If the folders specified in the configuration do not already exist, they are created.

Exports are different to other workflow outputs in this regard; subfolders cannot be defined using the Custom file name field. If slash characters are included in the Custom file name field for an export, all text before and including the final slash character is ignored.

9.1.7 Input

In addition to output, you also have to specify where the data should go into the workflow by adding an element called **Workflow Input**. This can be done by:

- Right-clicking the input box of the first tool and choosing **Connect to Workflow Input**. By dragging from the workflow input box to other input boxes several tools can use the input data directly.
- Pressing the button labeled **Add Element** (or right-click somewhere in the workflow background area and select **Add Element** from the menu that appears). The input box must then be connected to the relevant tool(s) in the workflow by dragging from the Workflow Input box to the "input description" part of the relevant tool(s) in the workflow.

At this point you have only prepared the workflow for receiving input data, but not specified which data to use as input.

Multiple input files can be used when:

- Data is generated within the Workflow
- Data is held within the Workbench
- Data is a combination of the two situations above

It can be useful to rename input elements when working with multiple input files, so that it is easy to discriminate between them when they are shown during workflow execution.

Note: Once the multiple input feature is used in a workflow, it is not possible to run the workflow in batch mode.

You can choose the order in which inputs will be processed by an element by right clicking on the input parameter box at the top of the element and choosing the option **Order Inputs**. This is most relevant for elements involved in data visualization. The feature **Order Inputs** is enabled when there are at least two inputs connected to the element (see figure 9.17). A small window will open, in which you can indicate the preferred order of the inputs to that element by moving them up and down in the list (figure 9.18). From this point forward, the order of the inputs is displayed on the branches connecting the inputs to elements.

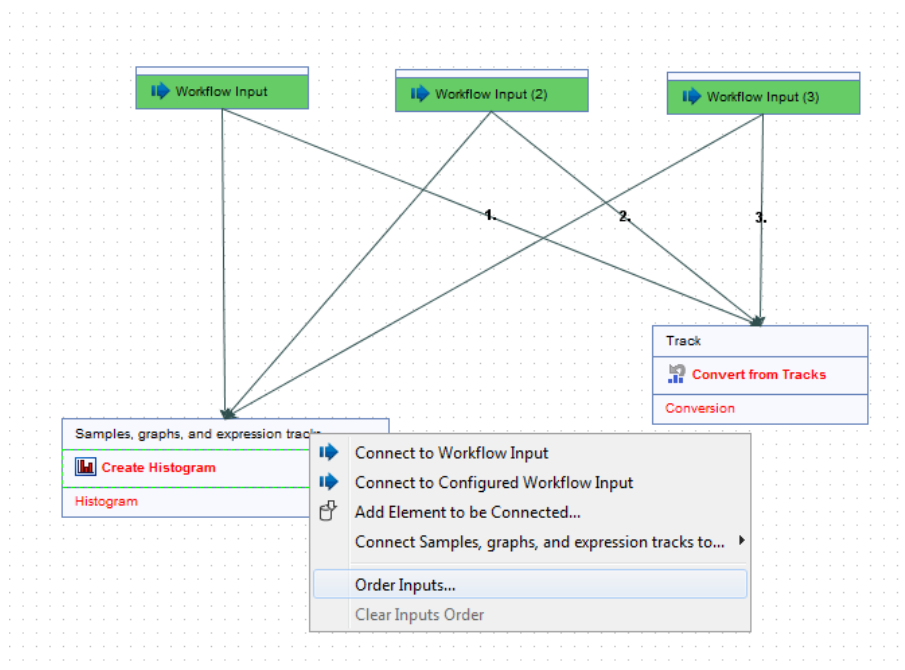


Figure 9.17: Right-click on the input parameter box to see the Order Inputs function.

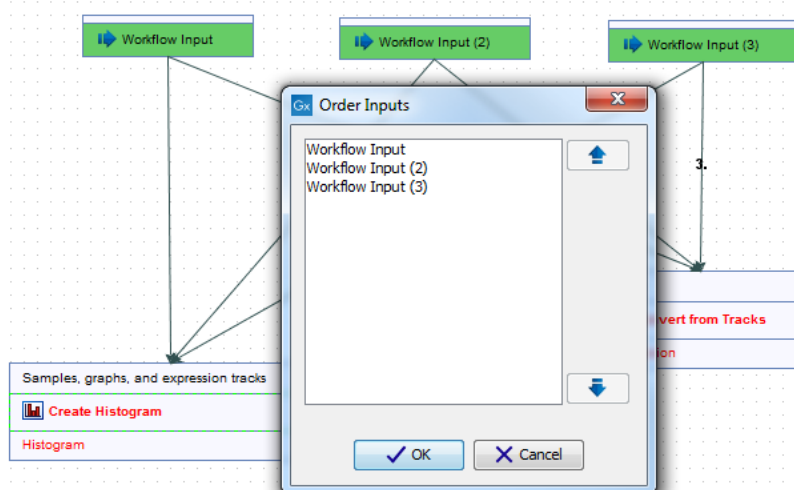


Figure 9.18: Define the inputs order for the element.

The feature **Order Workflow Inputs** allows you to set the order that a user will be asked for each input when they run the workflow. This option is enabled as soon as the workflow has two or more inputs (figure 9.19). Right click on empty space in the Workflow editor to start this tool. A small window will open in which the different inputs can be moved up and down to indicate the desired order (figure 9.20).

The example in figure 9.21 shows how to generate a track list in a workflow. Any track based on a compatible genome can be added to the same track list. This includes reference tracks as well as track results generated by elements of that workflow. In the latter case, only those for which

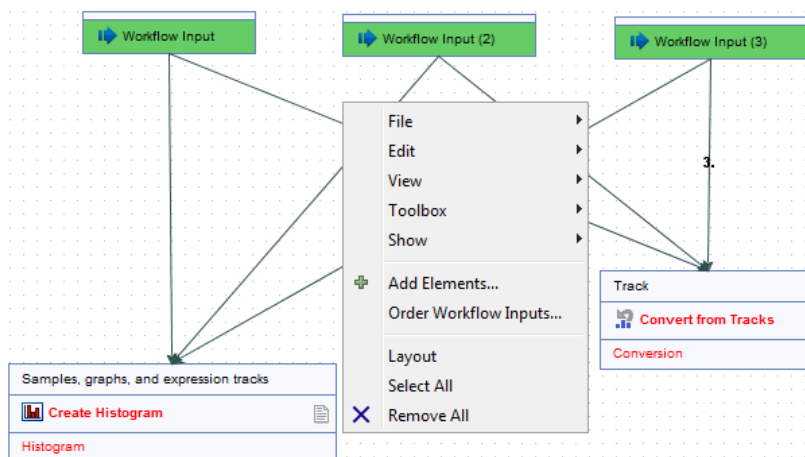


Figure 9.19: Right click on empty space in the Workflow editor to open the Order Workflow Inputs tool.

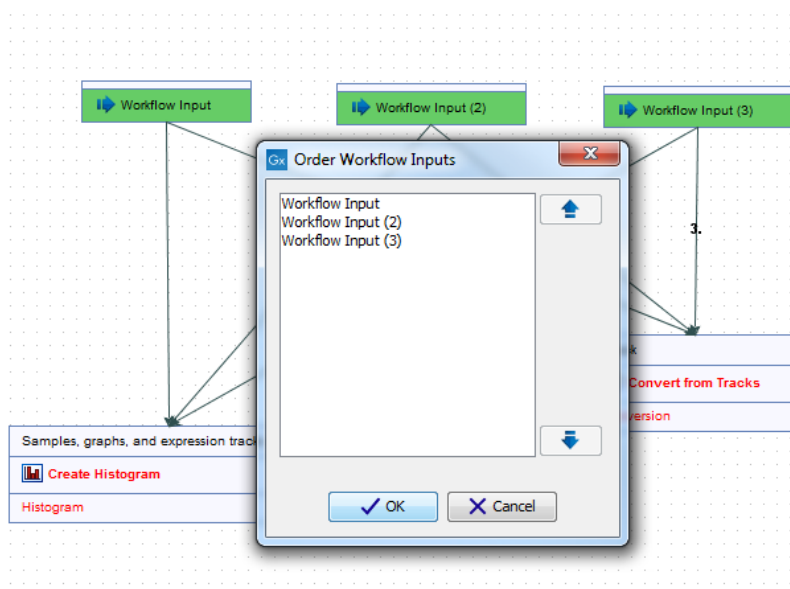


Figure 9.20: Define the order of the inputs for the workflow.

a workflow output element has been configured can be included in a track list.

9.1.8 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 9.22). 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

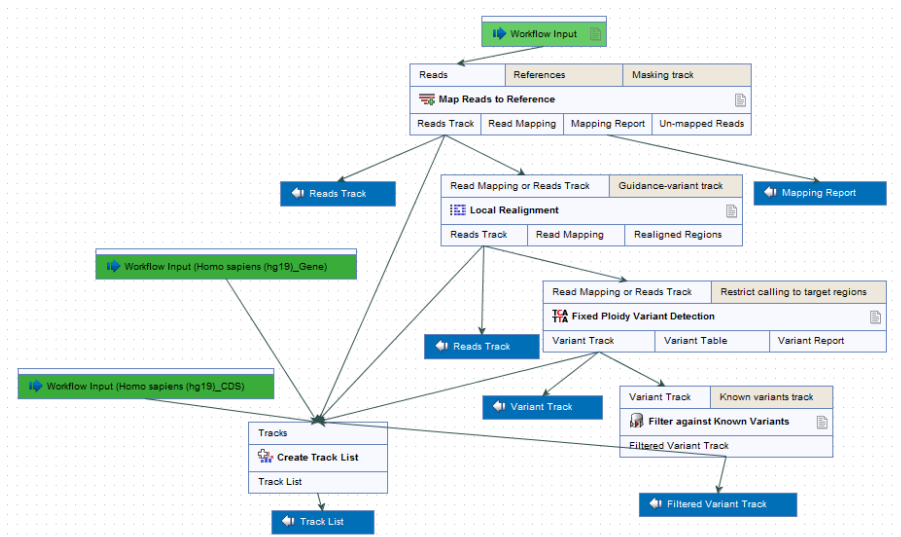


Figure 9.21: Generation of a track list including data generated within the Workflow, as well as data held in the Workbench.

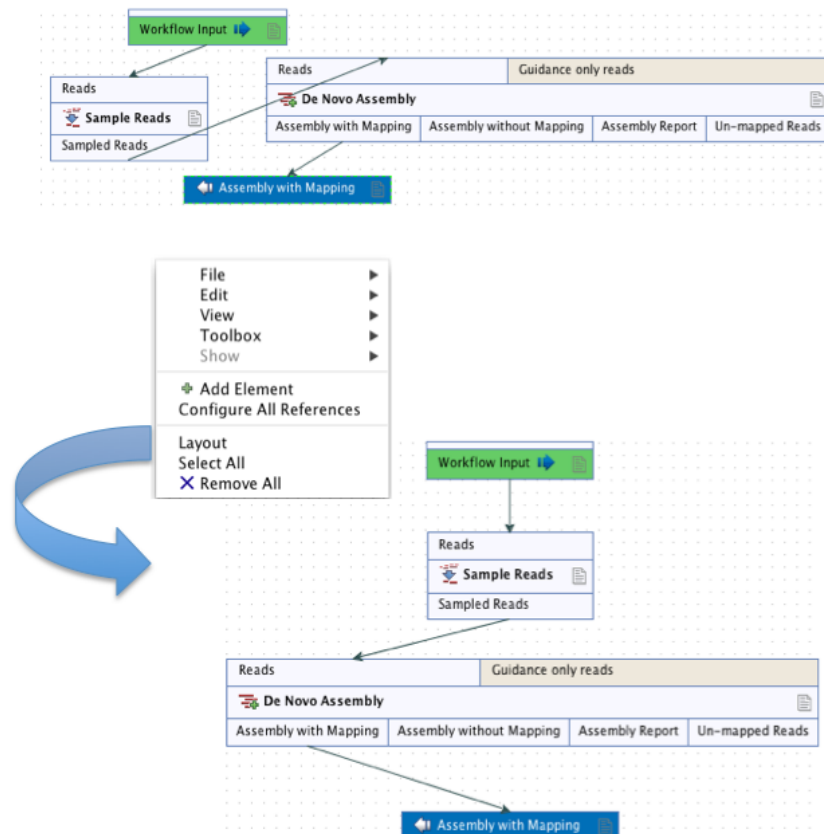



Figure 9.22: A workflow layout can be adjusted automatically with the "Layout" function.

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.

9.1.9 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 (M) (figure 9.23).



Figure 9.23: 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 9.24), 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 9.24).

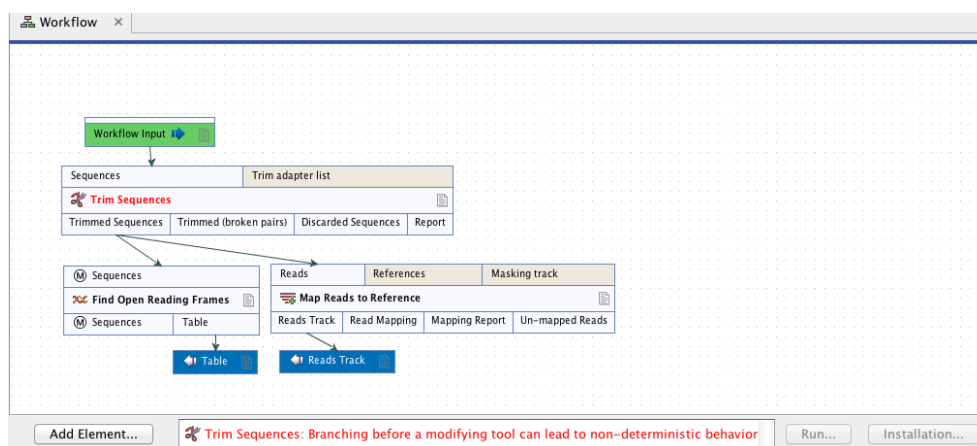


Figure 9.24: A branch containing an input modifying tool is not allowed in a workflow.

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 9.25 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".

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 9.26). A workflow output element can only be added when other tools than input modifying tools are included in the workflow.

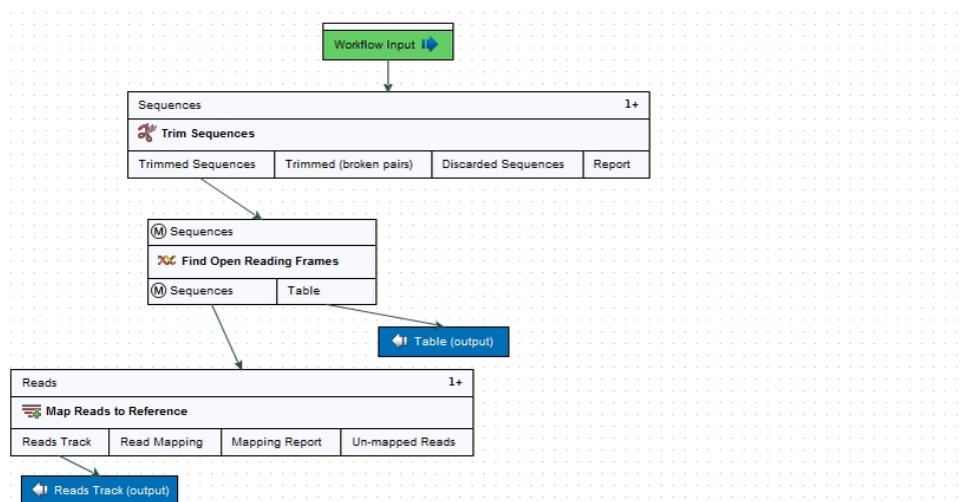


Figure 9.25: A branch containing an input modifying tool has been resolved and the workflow can now be run or installed.

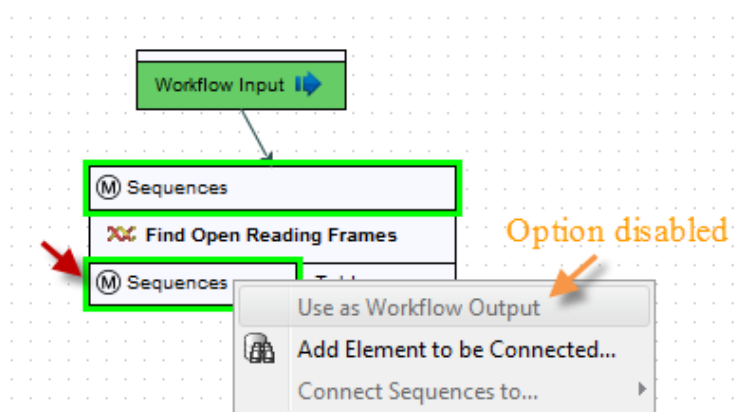


Figure 9.26: A workflow output element cannot be added if the workflow only contains an input modifying tool.

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 9.27). In order to see this output you must right click on the output option (marked with a red arrow in figure 9.27) and select "Use as Workflow Output".

When running a workflow where a workflow output has been added after the first input modifying tool in the chain (see figure 9.28) 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).

9.1.10 Workflow validation

At the bottom of the view, there is a text with a status of the workflow (see figure 9.29). It will inform about the actions you need to take to finalize the workflow.

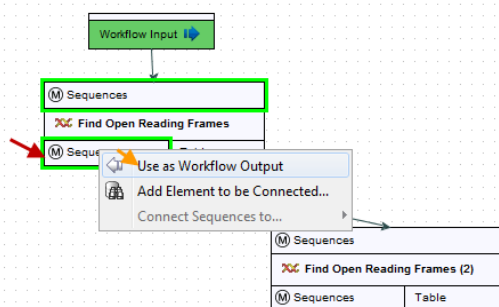


Figure 9.27: A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Select "Use as Workflow Output" to make a copy of the output.

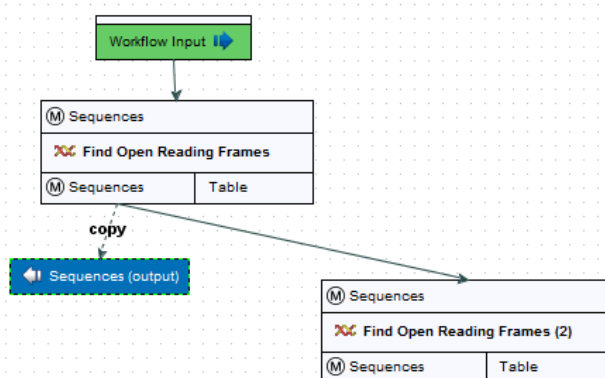


Figure 9.28: A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Note that this output is marked with "copy" to indicate that this is a copy of the result that is used as input at the next level in the chain.

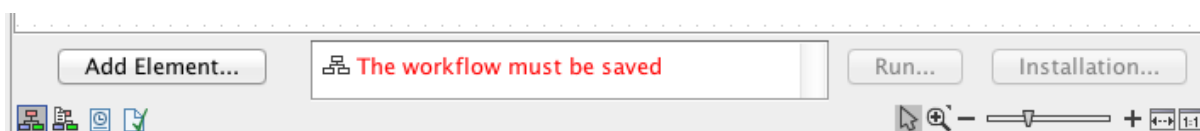


Figure 9.29: A workflow is constantly validated at the bottom of the view.

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.
- Additional checks that the workflow is consistent.

Once these conditions are fulfilled, the status will be "Validation successful". Clicking the **Run** 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 9.1.3), there will be a dialog asking for this as part of the test run.

9.1.11 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 9.30):

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.
- **Rulers** Vertical and horizontal rules can be visualised
- **Auto Layout** Ticking **Auto Layout** will ensure rearrangement of elements once new elements are added.
- **Connections to background** Connecting arrows are shown behind elements. This may ease reading of element names and accessible parameters.

Design

- **Round elements** Enable rounding of the element boxes.
- **Show shadow** Shadows of element boxes can be added.
- **Configured elements** Background color can be customized.
- **Input elements** Background color can be customized.
- **Edges** Color of connecting arrows can be customized.

9.1.12 Adding to workflows

Additional elements can be added to an already existing workflow by dragging it from the navigation area into the workflow editor and joining more elements as necessary. The new workflow must be saved and validated before it can be executed. Two or more workflows can be joined by dragging and dropping one from the Navigation Area, into another that is already open in the main viewing area. The output of one must be connected to the input of the next to allow the whole workflow to run in one go.

Workflows do not need to be valid to be dragged in to the workflow editor, but they must have been updated to the current version of the workbench.

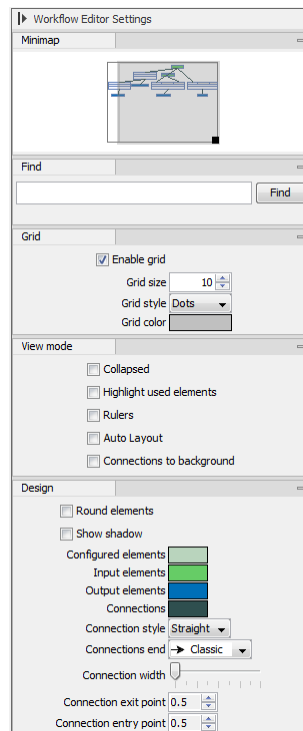


Figure 9.30: The Side Panel of the workflow editor.

9.1.13 Snippets in workflows

When creating a new workflow, you will often have a number of connected elements that are shared between workflows. Instead of building workflows from scratch it is possible to reuse components of an existing workflow. These components are called snippets and can exist of e.g. a read mapper and a variant detection tool.

Snippets can be created from an existing workflow by selecting the elements and the arrows connecting the selected elements. Next, you must right-click in the center of one of the selected elements. This will bring up the menu shown in figure 9.31.

When you have clicked on "Install as snippet" the dialog shown in figure 9.32 will appear. The dialog allows you to name the snippet and view the selected elements that are included in the snippet. You are also asked to specify whether or not you want to include the configuration of the selected elements and save it in the snippet or to only save the elements in their default configuration.

Click on the button labeled **OK**. This will install your snippet and the installed snippet will now appear in the **Side Panel** under the "Snippets" tab (see figure 9.33)

Right-clicking on the installed snippet in the **Side Panel** will bring up the following options (figure 9.34):

- **Add** Adds the snippet to the current open workflow
- **View** Opens a dialog showing the snippet, which allows you to see the structure
- **Rename** Allows renaming of the snippet.
- **Configure** Allows to change the configuration of the installed snippet.

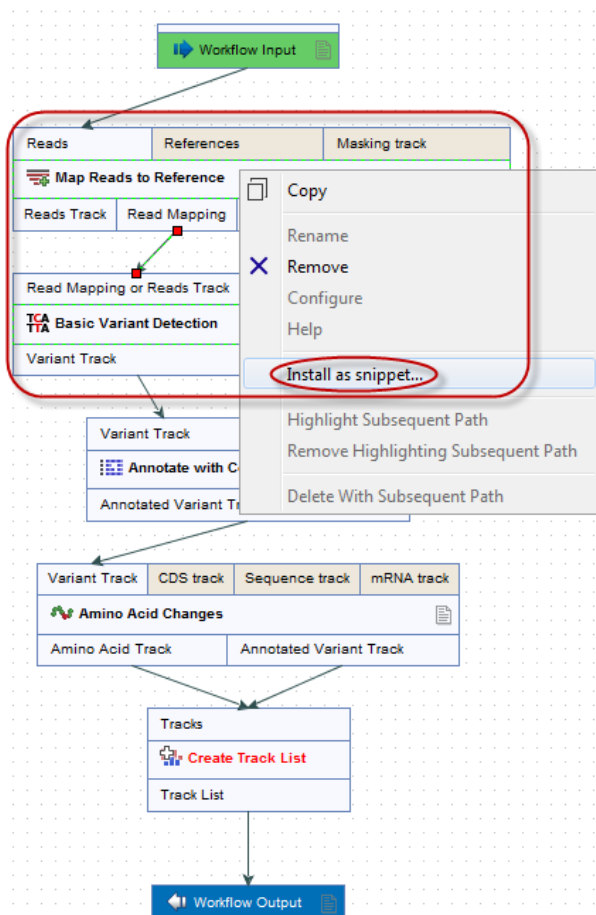


Figure 9.31: The selected elements are highlighted with a red box in this figure. Select "Install as snippet".

- **Uninstall** Removes the snippet.
- **Export** Exports the snippet to ones computer, allowing to share it.
- **Update** Updates the snippet (if update is required).

If you right-click on the top-level folder you get the options shown in figure 9.35:

- **Create new group** Creates a new folder under the selected folder.
- **Remove group** Removes the selected group (not available for the top-level folder)
- **Rename group** Renames the selected group (not available for the top-level folder)

In the **Side Panel** it is possible to drag and drop a snippet between groups to be able to rearrange and order the snippets as desired. An exported snippet can either be installed by clicking on the 'Install from file' button or by dragging and dropping the exported file directly into the folder where it should be installed.

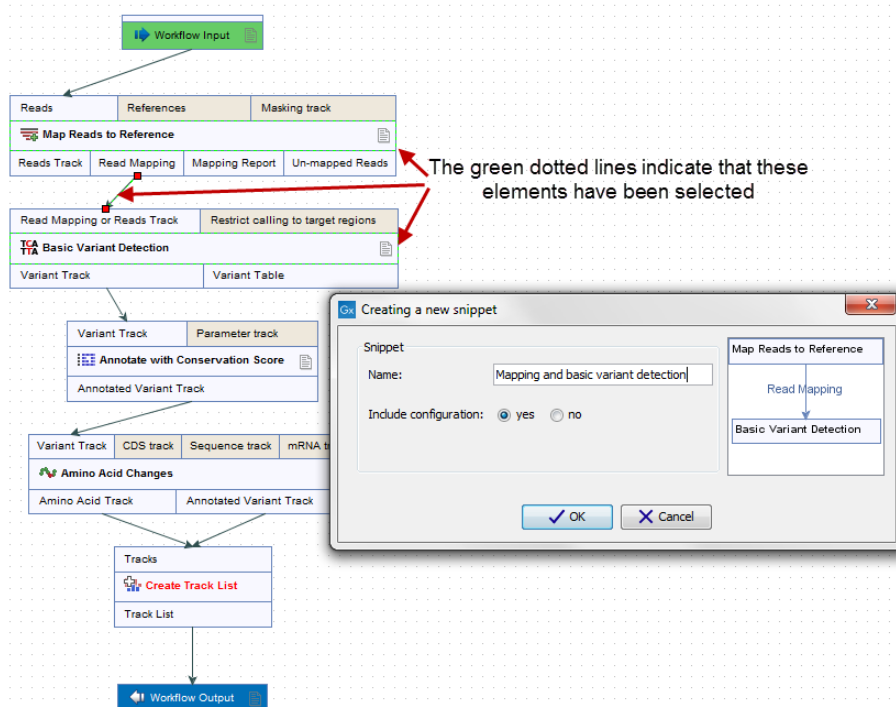


Figure 9.32: In the "Create a new snippet" dialog you can name the snippet and select whether or not you would like to include the configuration. In the right-hand side of the dialog you can see the elements that are included in the snippet.

Add a snippet to a workflow Snippets can be added to a workflow in two different ways; it can either be added by dragging and dropping the snippet from the **Side Panel** into the workflow editor, or it can be added by using the "Add elements" option that is shown in figure 9.36.

9.1.14 Change the order of tracks in the Track List

When modifying an existing workflow or creating a custom workflow that include the tool **Create Track List** you may want to be able to adjust the order in which the tracks are shown in the Track List. To do this, display a view of the workflow layout, click once on the top part of the tool Create Track List labeled "Tracks" followed by a right-click. In the pop up menu that appears (figure 9.37), choose the option "Order Workflow Inputs".

This opens a new pop up window (figure 9.38 where you can see a list of all the inputs that are connected with the input channel of the Create Track List tool. Use the arrows found in the left-hand side to move the tracks up or down until you have the desired track order in your Track List.

If the workflow also generated several variant tracks, the variant table generated from the uppermost read mapping will open in split view with the Track List. By changing the Order of Inputs you can thus also influence which variant table should open in split view.

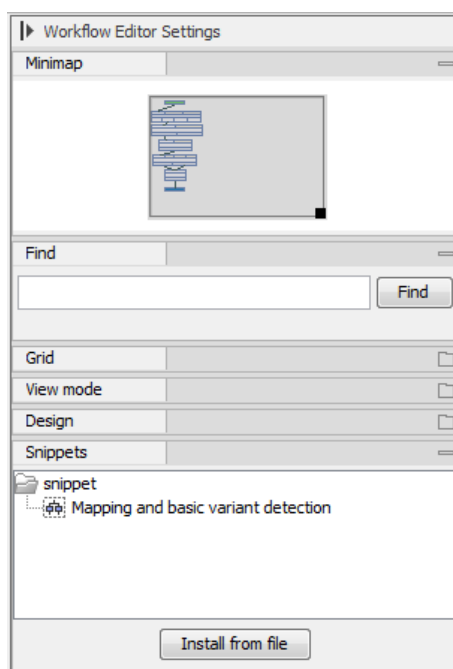


Figure 9.33: When a snippet is installed, it appears in the Side Panel under the "Snippets" tab.

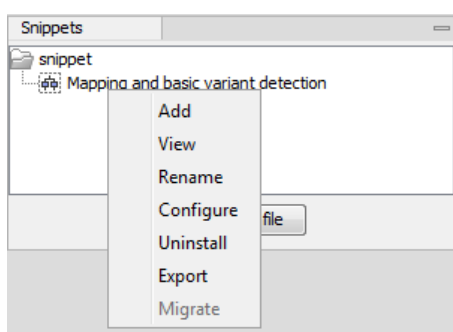


Figure 9.34: Right-clicking on an installed snippet brings up a range of different options.

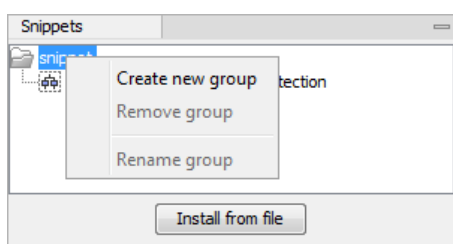


Figure 9.35: Right-clicking on the snippet top-level folder makes it possible to manipulate the groups.

9.2 Distributing and installing workflows

Once the workflow has been configured, you can use the **Run** button (see section 9.1.10) 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.

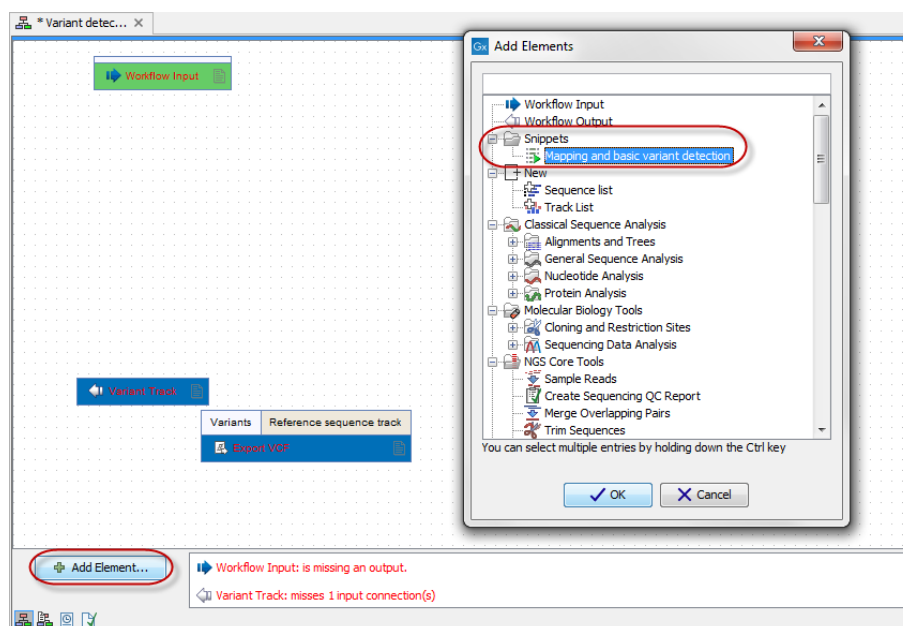


Figure 9.36: Snippets can be added to a workflow in the workflow editor using the 'Add Elements' button found in the lower left corner.

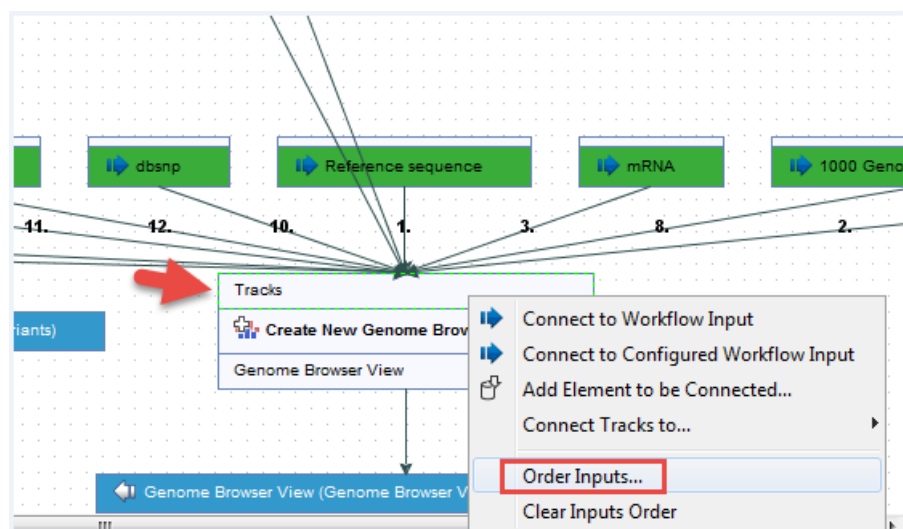


Figure 9.37: Right click on the workflow layout and choose the option "Order Inputs..."

9.2.1 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 in figure 9.39).

The information entered in this dialog will be visible for users installing the workflow and will enable them to look up the source of the workflow any time.

Author name Provide the name of the author of the workflow.

Author email Provide the email of the author of the workflow.

Author homepage Provide the homepage of the author of the workflow.

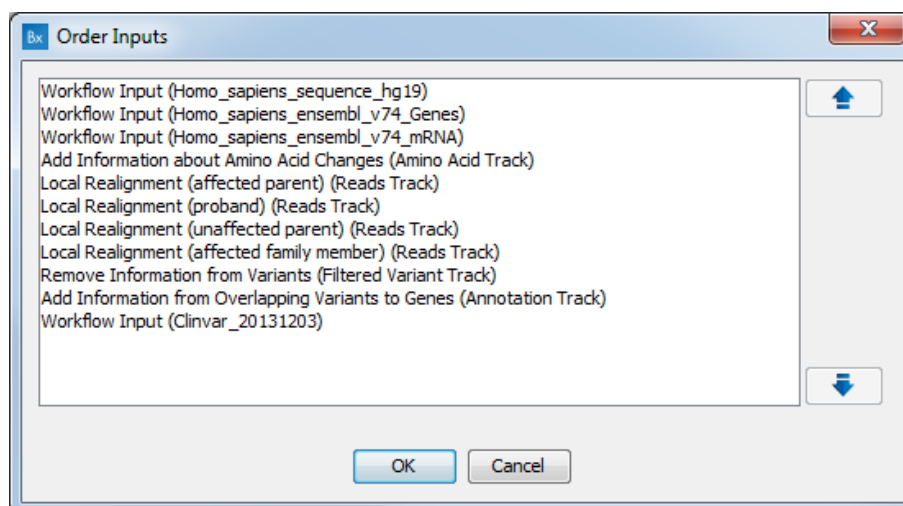


Figure 9.38: Example of Order Inputs window that appears when choosing the option "Order Inputs...".

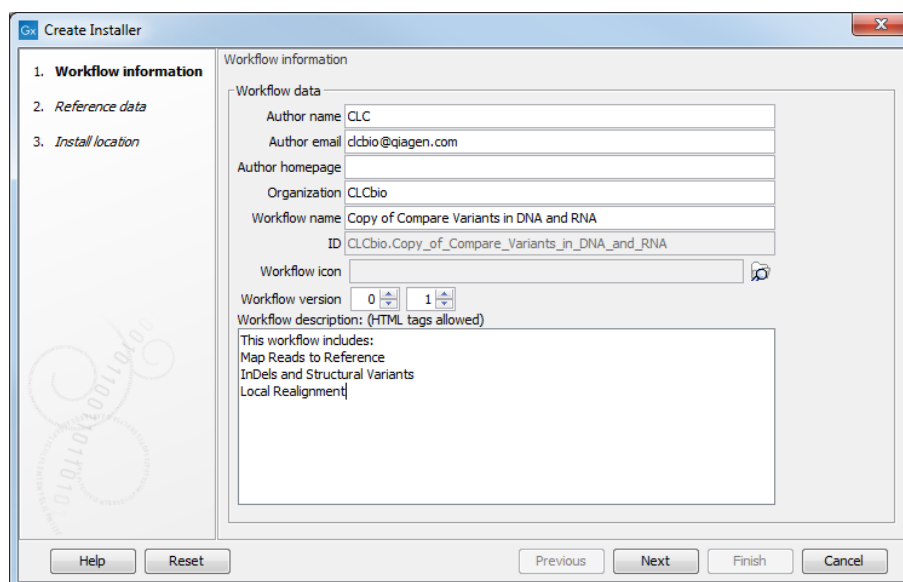


Figure 9.39: Workflow information for the installer.

Organization The organization name is important because it is part of the workflow id (see more in section 9.2.4).

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 9.2.4). 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.

Workflow description Provide a textual description of the workflow. This information will be displayed when a user mouses-over the name of the installed Workflow in the Workbench Toolbox, and is also presented in the Description tab for that Workflow in the Manage Workflows tool, described in section 9.2.3. Simple HTML tags are allowed (should be HTML 3.1 compatible, see <http://www.w3.org/TR/REC-html32>).

If you configured any of the workflow elements with data, clicking **Next** will give you the following options for the reference data (see figure 9.40). You can choose to

- **Ignore.** This is the recommended setting when the workflow inputs have been configured with workflow roles. The user installing the workflow on their local system will have to apply their own references.
- **Reference.** This is the recommended setting when the workflow inputs have been configured by selecting elements in a shared CLC_References directory. The data will not be bundled with the workflow, but the reference data is included in the workflow by pointing to the shared data in the CLC_References directory. This is particularly useful when working with large reference data.
- **Bundle.** This is the recommended setting when you cannot, or do not wish, to share the data through a CLC_References folder (see section ?? for how to transfer data to a CLC_References folder.) This option will include the data in the workflow by directly bundling the reference data with the workflow. **Note!** Bundling data should only be used to bundle small data sets with the workflow installer.

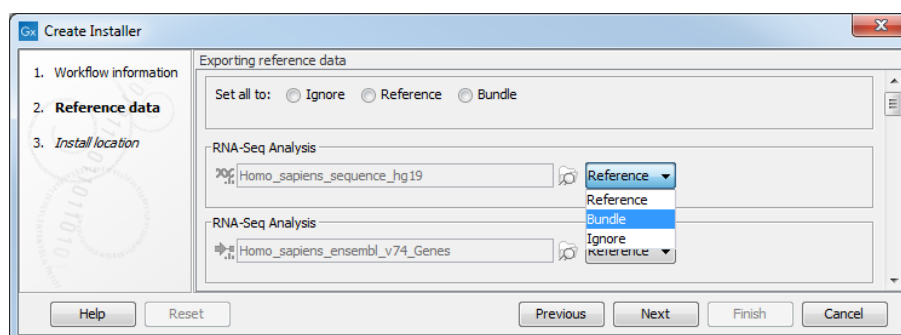


Figure 9.40: Bundling data with the workflow installer.

Click **Next** and you will be asked to specify where to install the workflow (figure 9.41). 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**. If you chose to bundle data with your workflow installation, you will be asked for a location to put the bundled data on the workbench. Installing a workflow with bundled data on a server, the data will be put in a folder created in the first writable persistence location. Should this location not suit your needs, you can always move it afterwards, using the normal persistence operations.

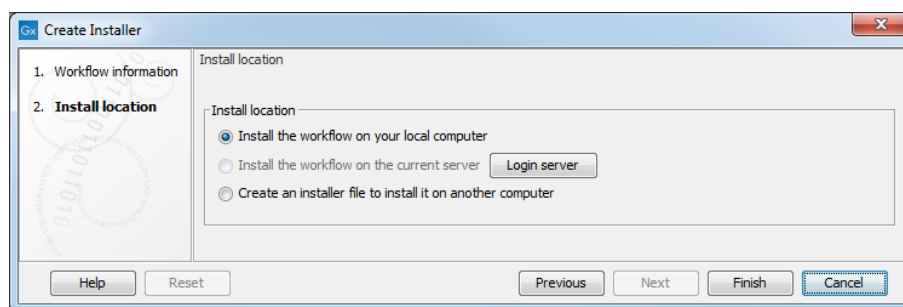


Figure 9.41: Select whether the workflow should be installed on your local computer or on the current server. A third option is to create an installer file (.cpw) that can be installed on another computer.

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 9.42) 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.

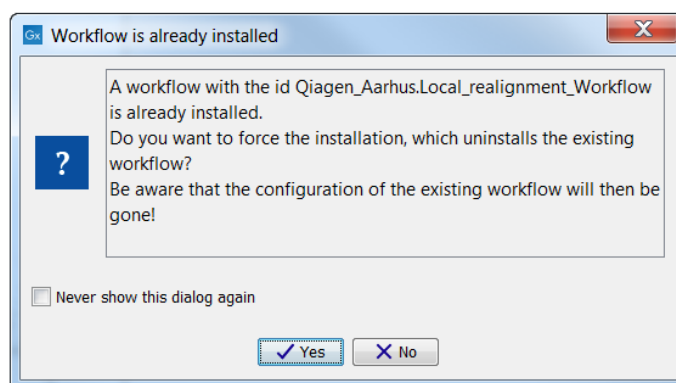


Figure 9.42: Select whether you wish to force the installation of the workflow or keep the original workflow.

9.2.2 Installing a workflow

Workflow .cpw files can be installed on a Workbench using the workflow manager:

Help | Manage Workflows (⚙️)

or press the "Workflows" button (🔧) in the toolbar and then select "Manage Workflow..." (⚙️).

To install a workflow, click on Install from File and select a .cpw file. If the workflow has bundled data, you will be prompted for a location for that data. Once installed, the workflow will appear under the Installed Workflows tab (figure 9.43).

Information on how to create a .cpw file can be found section 9.2.1.

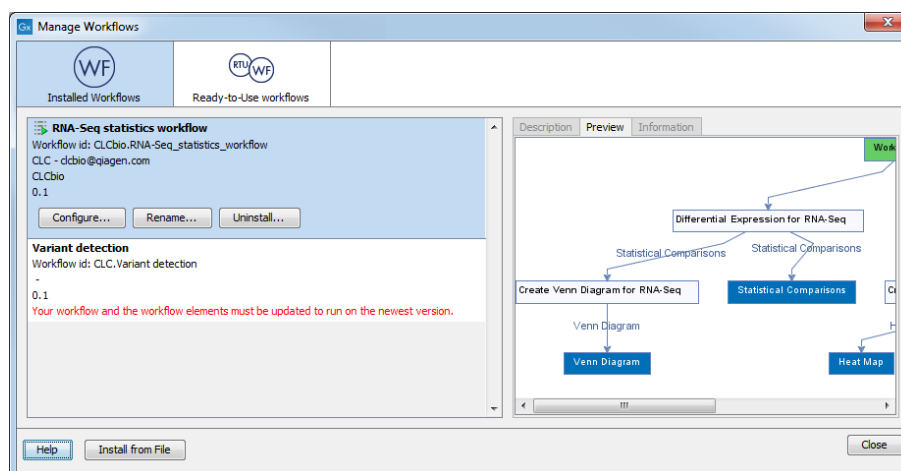


Figure 9.43: Workflows available in the workflow manager. Note the alert on the "Variant detection" workflow, that means that this workflow needs to be updated.

9.2.3 Managing workflows

Workflows can be managed from the workflow manager:

Help | Manage Workflows (⚙️)

or using the "Workflows" button (🔧) in the toolbar and then select "Manage Workflow..." (⚙️).

The workflow manager lists Installed workflows and Ready-to-Use workflows, but the functionalities described below (Configure, Rename, and Uninstall) are only available to custom workflows. You can always create a copy of a Ready-to-Use workflow (by opening the Ready-to-Use workflow and saving a copy in your Navigation Area) to enable the options described below.

Configure Select the workflow of interest and click on the button labeled Configure. You will be presented with a dialog listing all the reference data that need to be selected. An example is shown in figure 9.44.

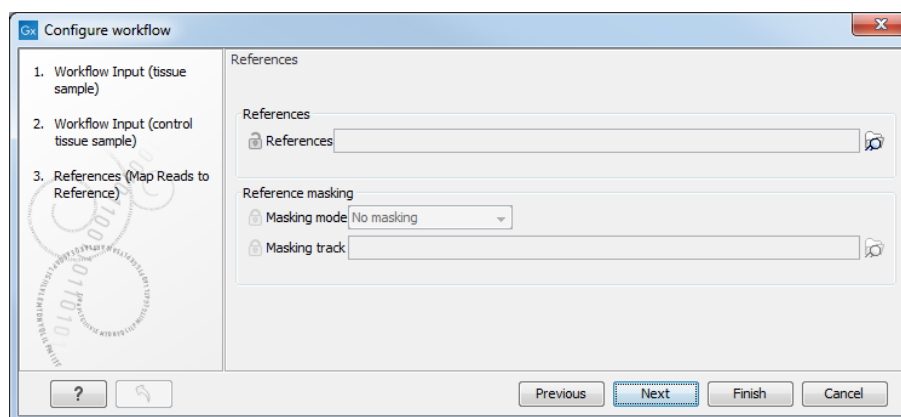


Figure 9.44: Configuring parameters for the workflow.

This dialog also allows you to lock parameters of the workflow (see more about locking in section 9.1.4).

Note that if the workflow is intended to be executed on a server, it is important to select reference

data that is located on the server.

Rename 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.

Uninstall Use this button to install a workflow.

Description, Preview and Information 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 9.39), the **Preview** shows a graphical representation of the workflow (figure 9.45), and finally you can get **Information** about the workflow (figure 9.46).

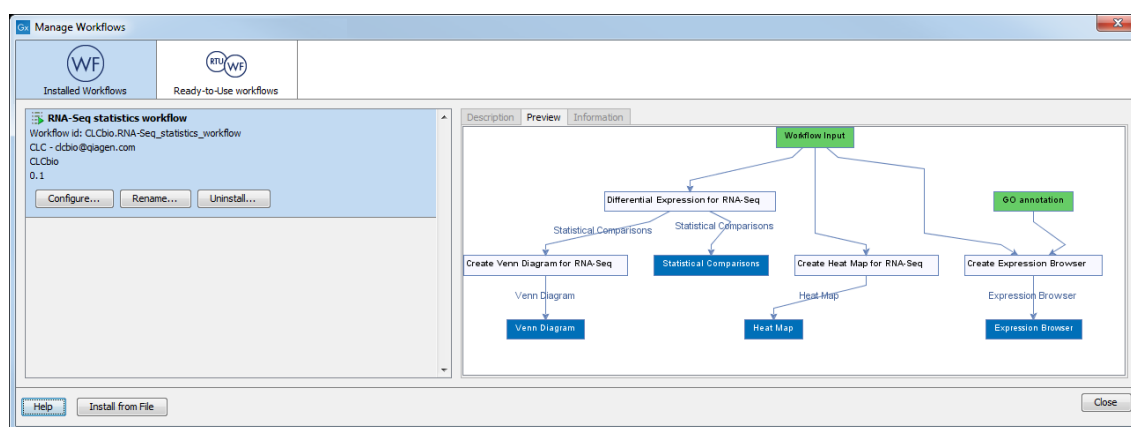


Figure 9.45: Preview of the workflow.

The "Information" field (figure 9.46) contains the following:

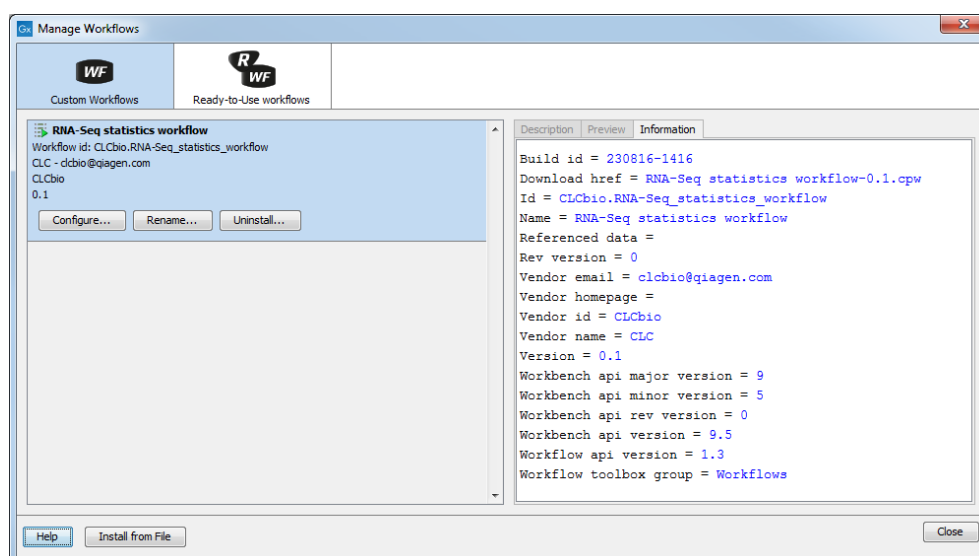


Figure 9.46: Workflow identification and versioning.

Build id The date (day month year) followed by the time (based on a 24 hour time) when the workflow was exported to a file through the Installation button at the bottom of the workflow window. If the workflow was installed locally without going through a file, the build ID will reflect the time of installation.

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)

9.2.4 Workflow version and update

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.

If a workflow needs to be updated, it must be updated before it can be used. Please note that:

- When you update a workflow, the older version is overwritten.
- If new parameters have been added to a tool as part of the update, these parameters will be set to their default values within the updated workflow.

Updating workflows in the Navigation area If a workflow stored in a CLC Data Location needs to be updated, this will become apparent when it is opened from the Navigation Area of the Workbench. In this case, an editor appears that lists the tools that need to be updated (figure 9.47). The workflow must be updated before it will be opened in the Workflow editor, and edited or launched. Click on the **OK** button at the bottom of the editor to update the workflow.

Updating installed workflows within the same major release line

"Major release line" refers to the first digit in the version number. For example: CLC Genomics Workbench 10.0.1 and 10.5 are releases within the same major release line. The major release number here is 10. CLC Genomics Workbench 9.x is part of a different major release line than 10.x because the major version number is different (9 versus 10).

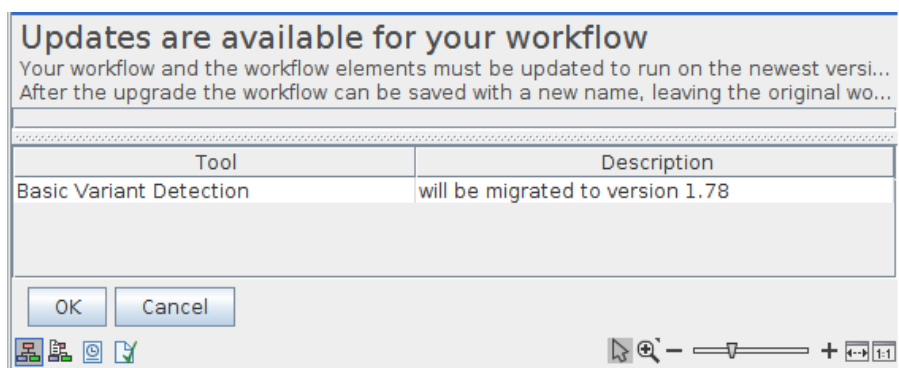


Figure 9.47: When updates are available, an editor appears with information about which tools should be updated. Press "OK" to update the workflow.

You can update installed workflows within the same major release line using the Workflow Manager tool.

Open the "Manage Workflows..." tool by selecting it after clicking on the "Workflows" button in the top toolbar of the Workbench (see figure 9.48) Select the workflow that needs to be updated. Workflows you installed directly will be under the "Custom Workflows" tab. Click on the "Update" button.

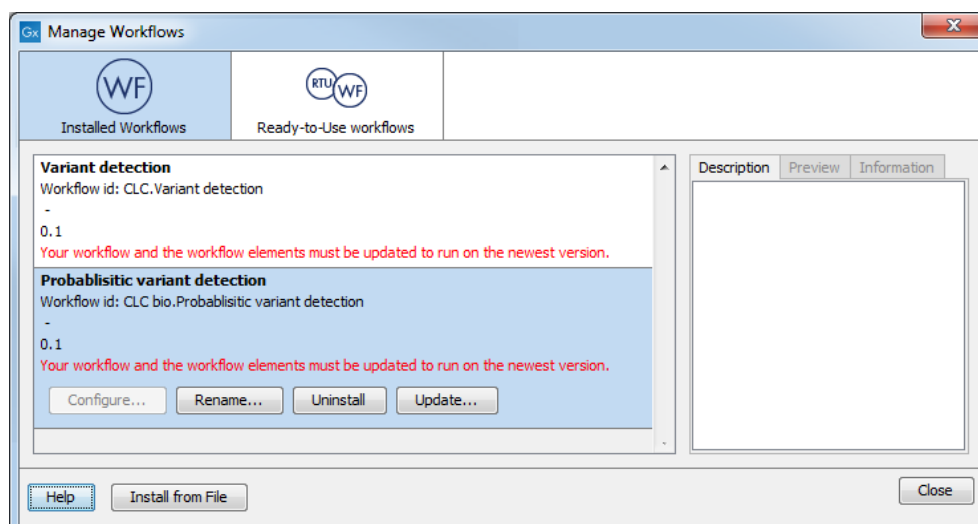


Figure 9.48: A message is shown indicating that a workflow needs to be updated. Clicking on that workflow selects it, and a button labeled "Update" will be visible.

To update a workflow you must have permission to write to the area the workflow is stored in. For workflows you installed directly, you will normally be able to do this when running the Workbench as you usually do. To update workflows distributed via plugins, it will usually mean running the Workbench as an administrative user.

Updating installed workflows between major release lines To update a workflow between major release lines, a Workbench version the installed workflow can be run on is needed, as well as the latest version of the Workbench.

To start, open a copy of the installed workflow in a version of the Workbench it can currently be run on. This is done by selecting the workflow in the **Installed Workflows** folder of the **Toolbox**

in the bottom left side of the Workbench, then right-clicking on the workflow name and choosing the option "Open Copy of Workflow" (figure 9.49).

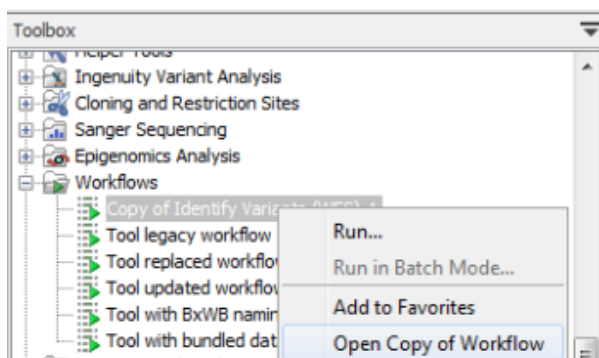


Figure 9.49: Open a copy of an installed workflow by right-clicking on its name in the Workbench Toolbox.

Save the copy of the workflow in the Navigation Area. To do this, you can simply drag and drop the tab to the location of your choice in the Navigation Area.

Open the new version of the Workbench and there, open the workflow that was saved in the Navigation Area. Click on the **OK** button if you are prompted to update the workflow.

You can now check that the workflow has been updated correctly, including that any reference data is configured as expected. Then save the updated version of the workflow. Finally, click the **Installation** button to install the workflow, if desired.

If the above process does not work when upgrading directly from a much older Workbench version, it may be necessary to upgrade step-wise by upgrading the workflow in sequentially higher major versions of the Workbench.

9.3 Executing a workflow

Once installed and configured, a workflow will appear in the **Toolbox** under **Installed Workflows** (📁). If an icon was provided with the workflow installer this will also be shown (see figure 9.50).

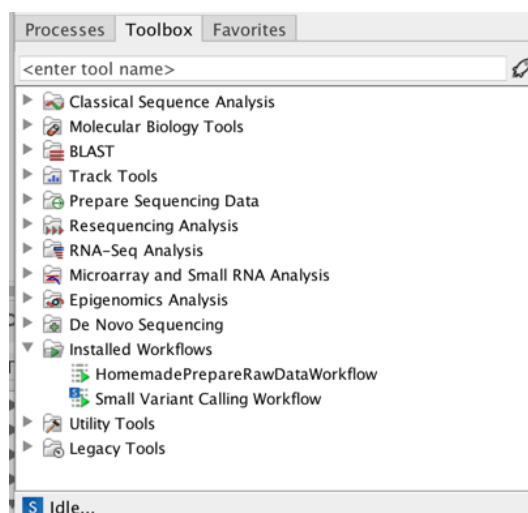


Figure 9.50: A workflow is installed and ready to be used.

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 select (and potentially download) relevant reference data, provide input data and configure the tools whose parameters have been left unlocked. Most workflows can also be run in batch mode (see section 8.3). In the last page of the dialog, you can preview all the parameters of the workflow, as well as the input data, before clicking "Next" to choose where to save the output, and then "Finish" to execute the workflow.

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 here: <http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/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 ¹.

9.4 Open copy of installed workflow

A copy of an installed and configured workflow found in the **Toolbox** under **Workflows** (📁) can be opened in the View Area by clicking once and then right-clicking on the name of the installed workflow in the toolbox (figure 9.51).

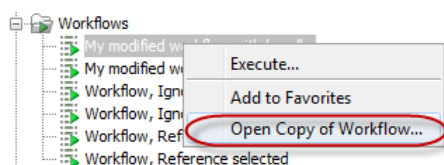


Figure 9.51: A copy of an installed workflow can be opened from the Toolbox. The copied workflow will open in the View Area.

An example of a copy of a workflow that has been opened in the **View Area** is shown in figure 9.52.

9.5 Batch launching workflows with multiple inputs

This section describes the launching of workflows with multiple inputs, where **all** input elements will be changed per batch. This launch mechanism is not intended for workflows with multiple input elements where one of the input elements remains the same in all batches, such as workflows meant to compare several tissues to a unique control tissue. At the moment, batch launching of such workflows is not possible, unless the common item is saved under different names as many times as there should be batches.

For workflows with multiple inputs where the inputs all need to change for each batch run, information specifying the grouping of the data elements and what role each element plays in a given analysis needs to be imported into the system from an Excel spreadsheet.

¹If 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 9.2.4)

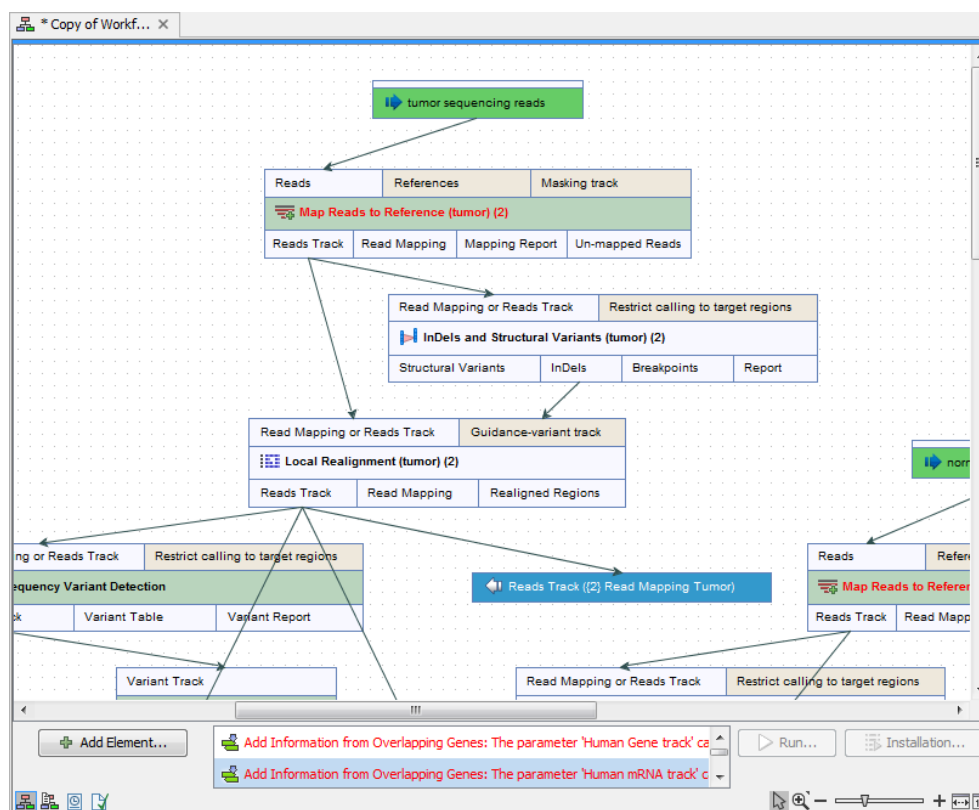


Figure 9.52: A copy of an installed workflow after it has been opened in the View Area.

The requirements for launching such workflows in batch mode are:

- The workflow must be installed on the Workbench, meaning that the workflow is accessible from the Toolbox (as opposed to workflows accessible from the Navigation Area). See section 9.2 to learn how to install a workflow.
- The workflow is characterized by more than one input file, and all input elements are unique per batch. You cannot reuse a common input element (such as control reads for example), unless it has been saved under different names in the Navigation Area.
- An Excel format file (.xlsx/.xls) must be provided, with at least 3 different columns:
 - **Unique ID** The first column must contain either the exact name of the data elements to be used as inputs, or partial name information such that data elements being entered into the analysis can be uniquely identified and matched with the information contained in the spreadsheet (see section 3.2.3 to learn more about matching partial names).
 - **Grouping** A second column must specify which data elements should be analyzed together in a given batch unit: this would be the ID of a single individual when comparing different tissues from the same individual (one individual per batch); or a family name when identifying variants existing within one family (one family per batch).
 - **Type** The third column must specify the type for each data element: the values in this column distinguish tissue samples from controls, or inform about the disease status of a family member (affected/non-affected/proband) when identifying disease causing variants.

(Figure 9.53) shows an example of a spreadsheet used in the case of tissue comparison. Note that the "grouping" and "type" are context specific, and will depend on the analysis performed, i.e., on the tools that constitute the workflow.

Unique ID = sample ID, exact of partial name of the reads file to ensure a unique match between reads and metadata.	Grouping = Identical values will be analyzed together in one batch unit, for example here Patient ID.	Type = value that defines which tissue is the control tissue and which is the sample tissue to be compared to the control.
23N	23	Normal
23T	23	Tumor
26N	26	Normal
26T	26	Tumor
27N	27	Normal
27T	27	Tumor
45N	45	Normal
45T	45	Tumor

Figure 9.53: Example of a spreadsheet necessary to run a workflow in batch, where the workflow intend to compare two tissue samples.

To launch a workflow with multiple input elements in batch mode, right click on the name of the workflow in the Toolbox and select the option "Run in Batch Mode..." (figure 9.54).

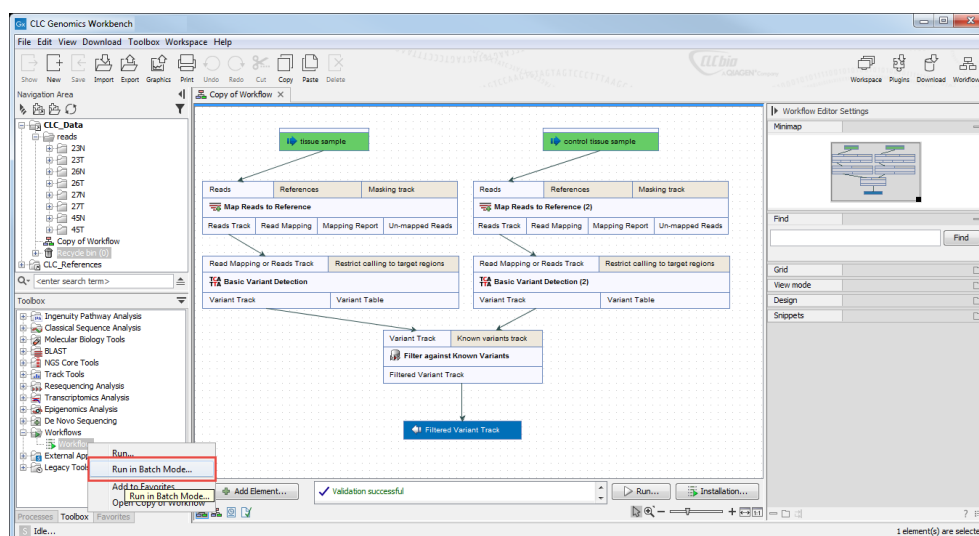


Figure 9.54: The option to "Run in Batch Mode..." appears in the context menu when you right click on the name of an installed workflow that has multiple input elements in the Toolbox panel.

A wizard opens and in the first window, you need to specify:

- An Excel file containing the information about the data to be analyzed (figure 9.55). Note that this file does not need to be saved in your Navigation Area. When it has been selected, the table found in the lower part of the wizard will show recapitulate the content of the Excel sheet. The location of the data for this analysis is not yet specified, so a red, no-entry sign is visible in the header of the first column.
- The location of the reads: click on the Navigation button next to the "Location of data" field and specify the folder(s) that contain(s) the data, as shown in figure 9.56.
Data elements within the selected folders are considered for the analysis. Subfolders and their contents are not considered unless the subfolder is also selected. Individual data elements cannot be selected.
- Select the appropriate matching scheme - exact or partial. The matching rules applied are the same as those used for metadata association: "Exact" means that data element

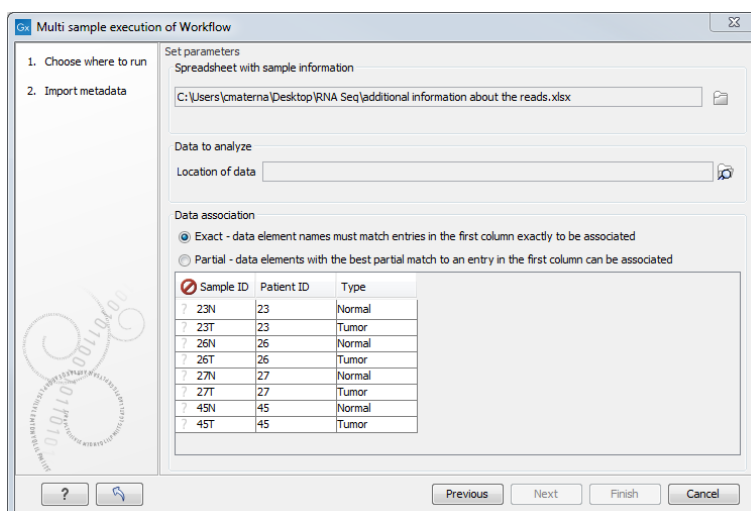


Figure 9.55: Select the information about the data to be analyzed and the folder holding the data to analyze. An example of an Excel sheet with the relevant information is shown.

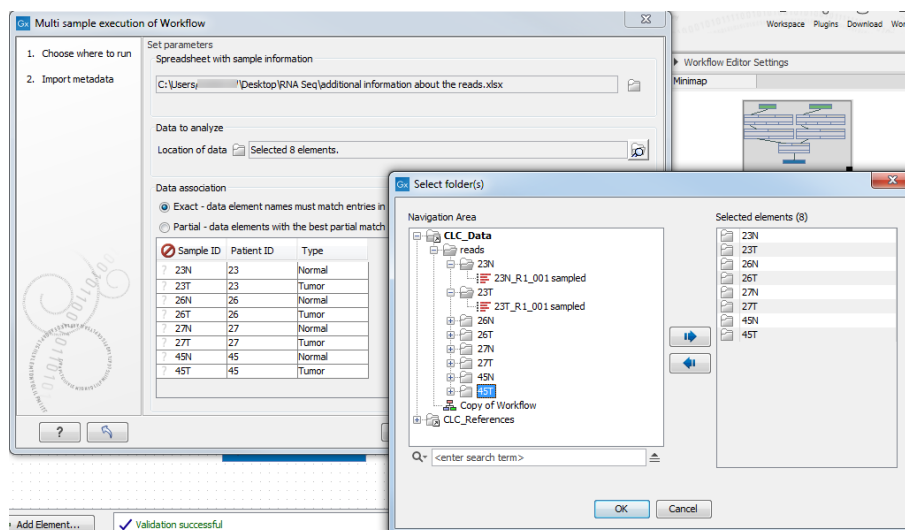


Figure 9.56: Select the folder(s) that contain(s) the data to be analyzed.

names must exactly match an entry in the first column of the Excel file; "Partial" matching allows for data elements names partially matching an entry in the first column. "Exact" is selected by default. Partial matching rules are described in detail in section 3.2.3.

An icon with a green check mark (✓) appears in the table preview next to rows where a data element corresponding to a row of the Excel sheet was uniquely identified. If no match can be made to a given row of the Excel sheet, a question mark (?) is displayed.

Graphical symbols are also presented in the header of the first column of the preview pane to give information about the overall status of the matching of rows in the Excel sheet with data elements in the Workbench:

- When no data elements match information in the Excel sheet, a red, no entry symbol (⊘) is displayed. In this situation, the button labeled **Next** is not enabled. This is the expected state before any data elements have been selected.

- A yellow exclamation mark (⚠) indicates that some, but not all rows in the Excel sheet have been matched to a data element in the selected folder(s).
- A green checkmark (✓) indicates that all rows in the Excel sheet have been matched to a data element in the selected folder(s).

In figure 9.57, the green check mark symbol in the header of the first column in the preview pane indicates that data elements were identified for each of the rows in the Excel sheet. You can click on the button labeled "Next".

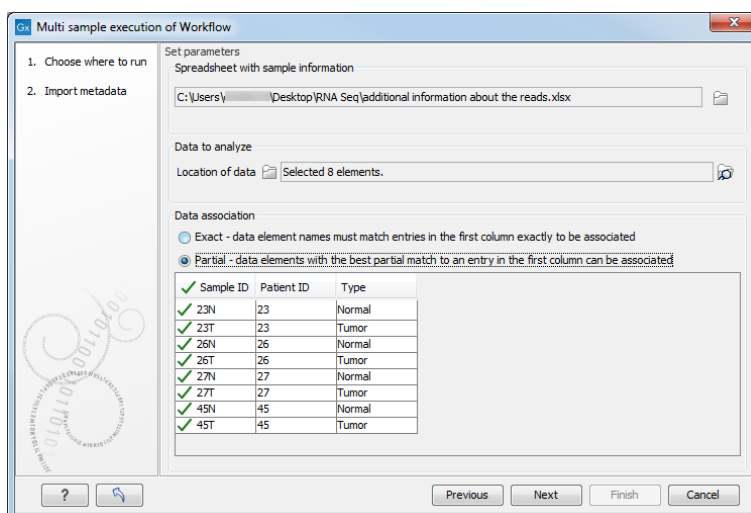


Figure 9.57: View of the Data Association table after all samples were successfully associated.

The next wizard window is called "Select grouping parameters and analysis inputs".

- In the **Group by** drop down menu, select the name of the column containing information that specifies which samples should be analyzed together.
- In the **Type** drop down menu, select the name of the column containing information that can be mapped to the workflow input type of each data element.

In the same window you will need to further specify the inputs of the workflow. What needs to be specified here is dependant on the workflow itself.

An example is shown in figure 9.58. **Group by** is set to a column specifying "Patient ID", because each workflow run will analyze a sample pair. **Type** is set to the "Type" column, because the workflow inputs are either tumor or normal tissues. The sample columns section maps data elements to the different workflow inputs, in this case "Tissue sample" is set to "Tumor", and "Control tissue sample" to "Normal".

The rest of the wizard is dependant of the tools included in the workflow. Fill in the appropriate information and save the results of your workflow in a folder you can create in the Navigation Area.

As in a regular batching mode, you can use the progress bar to see how the job is progressing (figure 9.59): a process called "Batch Process" indicates how many batches have been completed, while the ones situated above show the analysis progress of a particular batch unit.

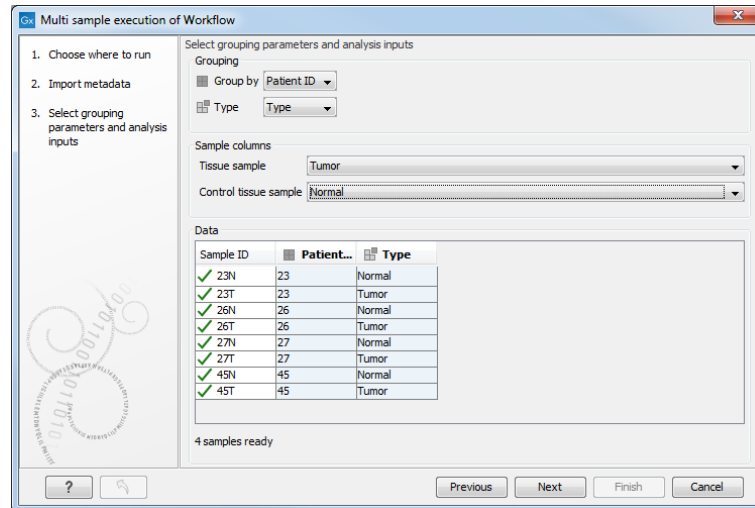


Figure 9.58: Grouping samples.

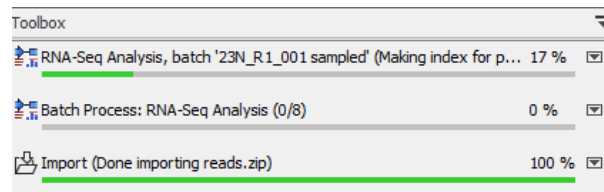


Figure 9.59: Check on the progress of your workflow being run in batch mode using the Processes tab below the Toolbox.

Chapter 10

Other data types

Contents

10.1 Tracks	189
--------------------------	------------

10.1 Tracks

The *CLC Main Workbench* supports viewing of data in track format, which is the preferred data format used to visualize and analyze data in the CLC Genomics Workbench. For a description of the track format we refer to <http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tracks.html>.

Part III

Bioinformatics

Chapter 11

Viewing and editing sequences

Contents

11.1 View sequence	191
11.1.1 Sequence settings in Side Panel	192
11.1.2 Selecting parts of the sequence	198
11.1.3 Editing the sequence	199
11.1.4 Sequence region types	200
11.2 Circular DNA	201
11.2.1 Using split views to see details of the circular molecule	201
11.2.2 Mark molecule as circular and specify starting point	202
11.3 Working with annotations	203
11.3.1 Viewing annotations	204
11.3.2 Adding annotations	208
11.3.3 Edit annotations	210
11.3.4 Removing annotations	211
11.4 Element information	212
11.5 View as text	213
11.6 Sequence Lists	214

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.

11.1 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 2.2 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 13.2).

11.1.1 Sequence settings in Side Panel

Each view of a sequence has a **Side Panel** located at the right side of the view (see figure 11.1).

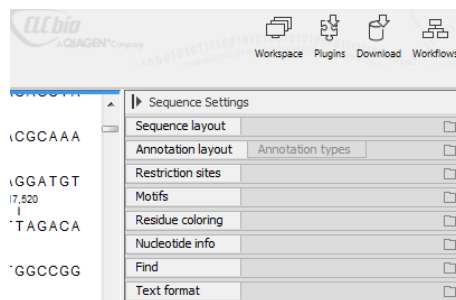



Figure 11.1: Overview of the Side Panel which is always shown to the right of a view.

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 4.6 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 [11.3.1](#).

Restriction sites

See section [11.1.1](#).

Motifs

See section [15.9.1](#).

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.

- **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. In cases where variants are present in the reads, synonymous variants are shown in orange in the translated sequence whereas non-synonymous are shown in red.
 - **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 [11.1.2](#).

- * **+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 16.5.
- **Only AUG start codons.** For most genetic codes, a number of codons can be start codons (TTG, CTG, or ATG). These will be colored green, unless selecting the "Only AUG start codons" option, which will result in only the AUG codons colored in green.
- **Single letter codes.** Choose to represent the amino acids with a single letter instead of three letters.
- **Trace data.** See section 18.1.
- **Quality scores.** For sequencing data containing quality scores, the quality score information can be displayed along the sequence.
 - **Show as probabilities.** Converts quality scores to error probabilities on a 0-1 scale, i.e. not log-transformed.
 - **Foreground color.** Colors the letter using a gradient, where the left side color is used for low quality and the right side color is used for high quality. The sliders just above the gradient color box can be dragged to highlight relevant levels. The colors can be changed by clicking the box. This will show a list of gradients to choose from.
 - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
 - **Graph.** The quality score is displayed on a graph (Learn how to export the data behind the graph in section 6.4).
 - * **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 6.4).
 - * **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 21.2.3 for a detailed description of the settings.

Protein info

These preferences only apply to proteins. The first nine items are different hydrophobicity scales. These are described in section 17.3.1.

- **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. The option "Include translations" means that you can choose to search for translations *which are part of*

an annotation (in some cases, CDS annotations contain the amino acid sequence in a "/translation" field). But it will not dynamically translate nucleotide sequences, nor will it search the translations that can be enabled using the "Nucleotide info" side panel.

- **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 and end number. 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 and mapping results for example.

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.

Restriction sites in the Side Panel

Please see section [20.1.1](#).

11.1.2 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 [11.1.1.](#))

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.

11.1.3 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 correct only one residue, this is possible by simply making the selection cover only one residue and then type the new residue.

Another way to edit the sequence is by inserting a restriction site. See section [20.3.4](#).

Note When editing annotated nucleotide sequences, the annotation content is not updated automatically (but its position is). Please refer to section 11.3.3 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.

11.1.4 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 11.2 is an example of three regions with separate colors.

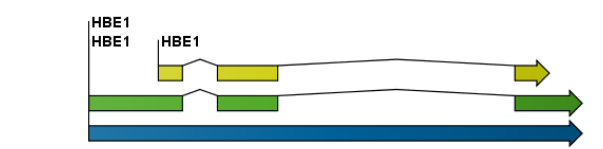


Figure 11.2: Three regions on a human beta globin DNA sequence (HUMHBB).

Figure 11.3 shows an artificial sequence with all the different kinds of regions.

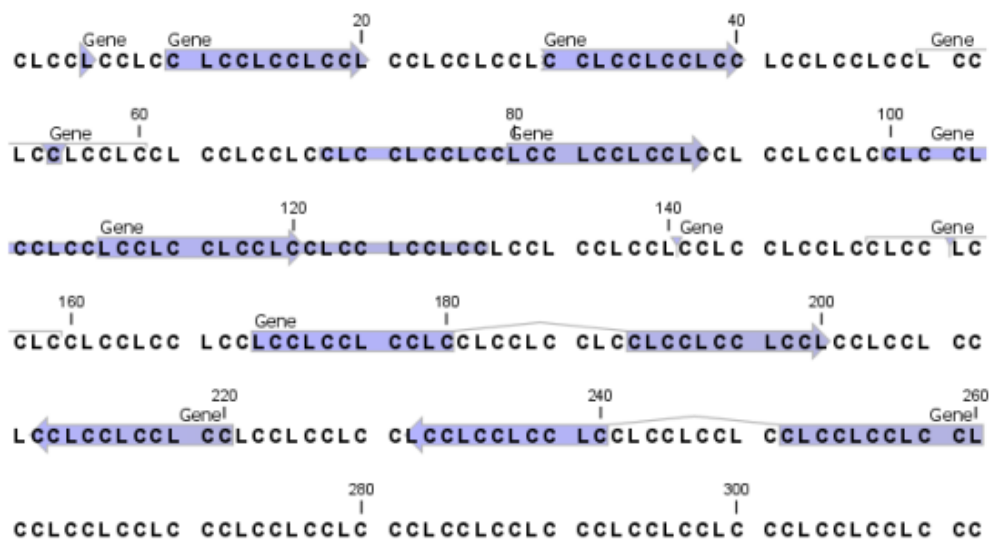


Figure 11.3: Region #1: A single residue, Region #2: A range of residues including both endpoints, Region #3: A range of residues starting somewhere before 30 and continuing up to and including 40, Region #4: A single residue somewhere between 50 and 60 inclusive, Region #5: A range of residues beginning 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.

11.2 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" ()

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 11.4.

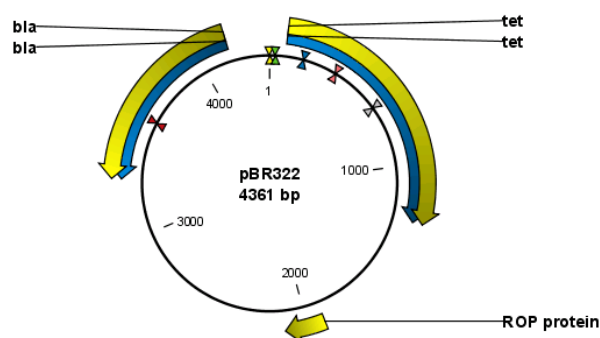


Figure 11.4: A molecule shown in a circular view.

This view of the sequence shares some of the properties of the linear view of sequences as described in section 11.1, 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.

11.2.1 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 () at the bottom of the view

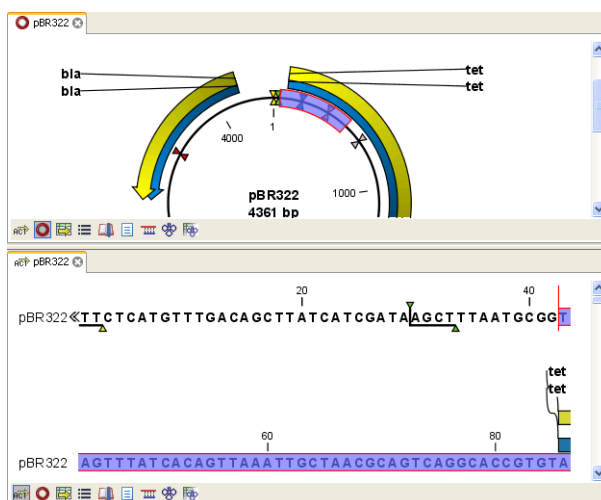


Figure 11.5: Two views showing the same sequence. The bottom view is zoomed in.

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 11.5.

Note! If you make a selection in one of the views, the other view will also make the corresponding selection, providing an easy way for you to focus on the same region in both views.

11.2.2 Mark molecule as circular and specify starting point

You can mark a DNA molecule as circular or linear by right-clicking on its name in either the Sequence view or the Circular view. If the sequence is linear, you will see the option to mark it as circular and vice versa (see figure 11.6).

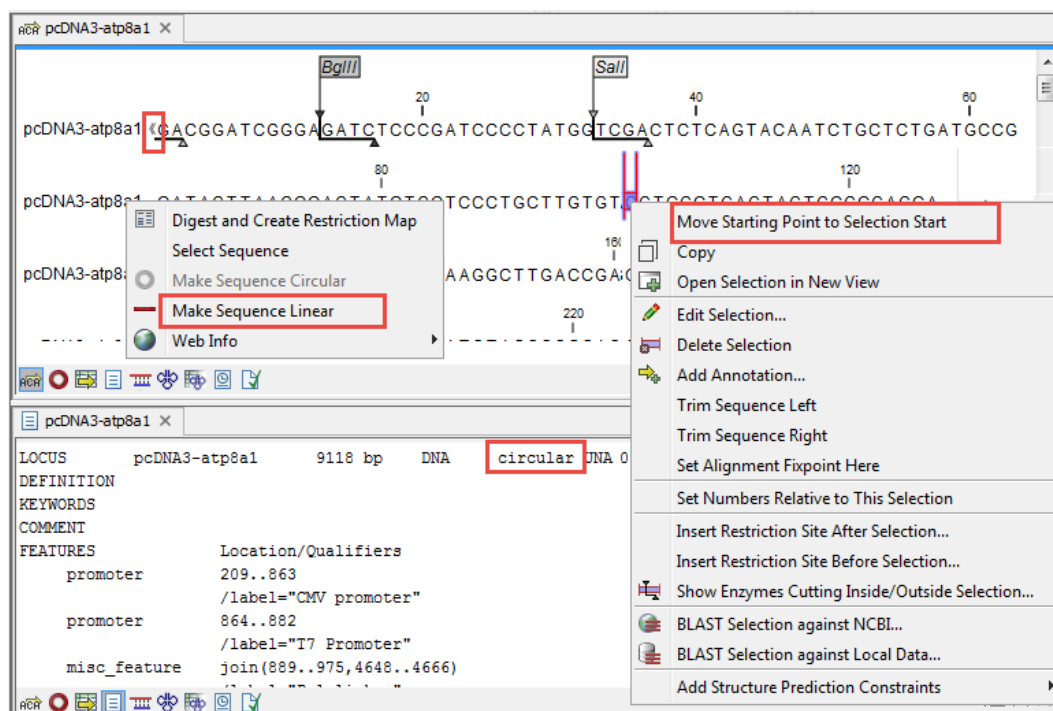


Figure 11.6: Double angle brackets marks the start and end of a circular sequence seen in linear view. Below, the Text view of the same sequence shows the mention circular in the first line.

In the Sequence view, a sequence marked as circular is indicated by the use of double angle brackets at the start and end of the sequence. The linear or circular status of a sequence can also be seen in the Locus line of the Text view for a Sequence, or in the Linear column of the Table view of a Sequence List.

The starting point of a circular sequence can be changed by selecting the position of the new starting point and right-clicking on that selection to choose the option **Move Starting Point to Selection Start** (figure 11.7).

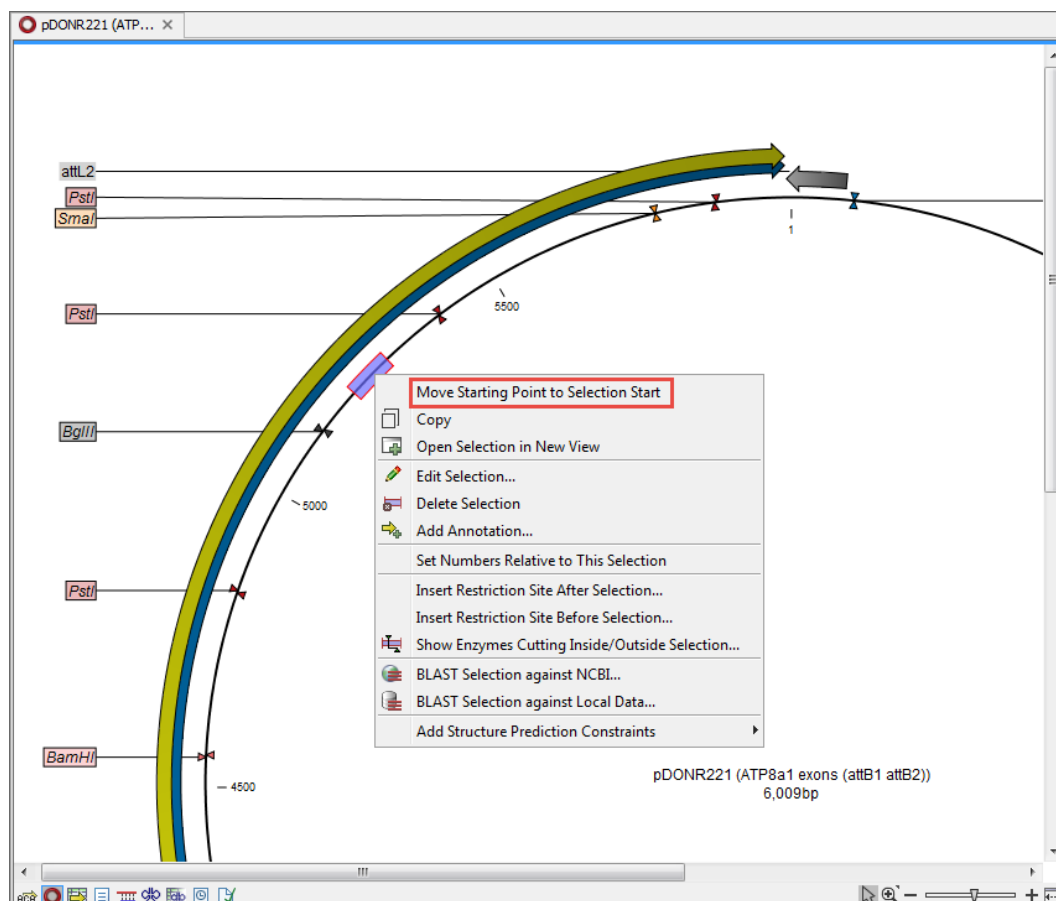


Figure 11.7: Right-click on a circular sequence to move the starting point to the selected position.

11.3 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).
- A protein structure can be linked with a sequence (section 12.4.2), and atom groups defined on the structure transferred to sequence annotations or vica versa (section 12.4.3).
- You can manually add annotations to a sequence (described in the section 11.3.2).

If you would like to extract parts of a sequence (or several sequences) based on its annotations, you can find a description of how to do this in section 24.2.

Note! Annotations are included if you export the sequence in GenBank, Swiss-Prot, EMBL or CLC format. When exporting in other formats, annotations are not preserved in the exported file.

11.3.1 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 11.8 shows an annotation displayed on a sequence.

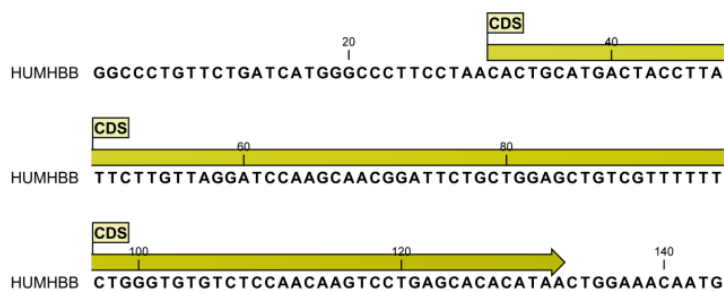


Figure 11.8: An annotation showing a coding region on a genomic dna sequence.

The various sequence views listed in section 11.3.1 have different default settings for showing annotations. However, they all have two groups in the **Side Panel** in common:

- **Annotation Layout**
- **Annotation Types**

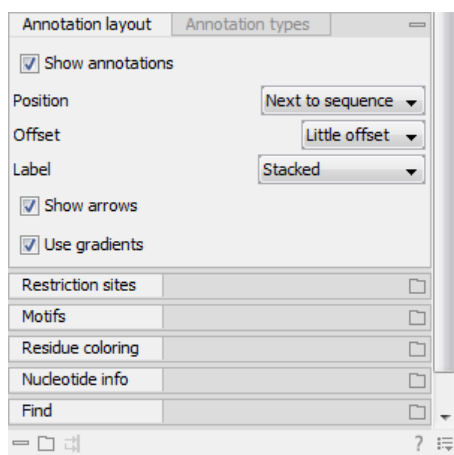


Figure 11.9: The annotation layout in the Side Panel. The annotation types can be shown by clicking on the "Annotation types" tab.

The two groups are shown in figure 11.9.

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 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 five tabs: Swatches, HSB, HSI, RGB, and CMYK. They represent five 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 11.10).

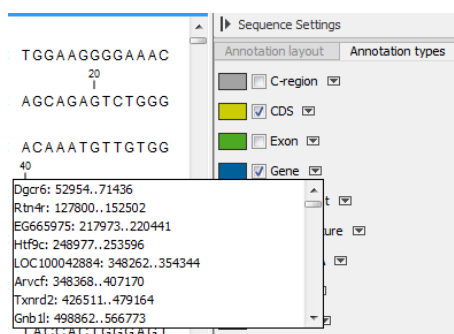


Figure 11.10: Browsing the gene annotations on a sequence.

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.

Note: A waved end on an annotation (figure 11.11) means that the annotation is torn, i.e., it extends beyond the sequence displayed. An annotation can be torn when a new, smaller sequence has been created from a larger sequence. A common example of this situation is when you select a section of a stand-alone sequence and open it in a new view. If there are annotations present within this selected region that extend beyond the selection, then the selected sequence shown in the new view will exhibit these torn annotations.

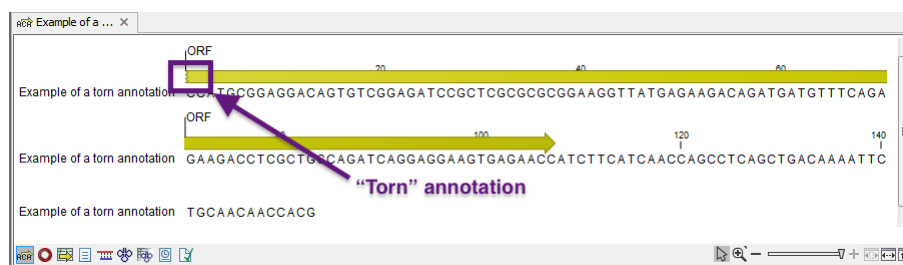


Figure 11.11: Example of a torn annotation on a sequence.

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 11.12).

Name	Type	Region	Qualifiers
Atp8a1	Gene	1..228194	/gene=Atp8a1 /note=Derived by automated computational analysis using gene prediction method: BestRefseq. Supporting evidence includes similarity to: 2 mRNAs /db_xref="GeneID:11980" /db_xref=MGI:1330848
Atp8a1	CDS	join(222..270,32851..32...	/gene=Atp8a1 /GO_component=integral to membrane; membrane /GO_function=ATP binding; ATPase activity; ATPase activity, coupled to transmembrane movement of ions, phosphorylative mechanism; catalytic activity; hydrolase activity; hydrolase activity, acting on acid anhydrides, catalyzing transmembrane movement of substances; magnesium ion binding; metal ion binding; nucleotide binding; phospholipid-translocating ATPase activity /GO_process=cation transport; metabolism /note=isoform b is encoded by transcript variant 2; ATPase 8A1, p...

Figure 11.12: A table showing annotations on the sequence.

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.

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 11.3.2).

You can benefit from this table in several ways:

- It provides an intelligible overview of all the annotations on the sequence.
- You can use the filter at the top to search the annotations. Type e.g. "UCP" into the filter and you will find all annotations which have "UCP" in either the name, the type, the region or the qualifiers. Combined with showing or hiding the annotation types in the **Side Panel**, this makes it easy to find annotations or a subset of annotations.
- You can copy and paste annotations, e.g. from one sequence to another.
- If you wish to edit many annotations consecutively, the double-click editing makes this very fast (see section 11.3.2).

11.3.2 Adding annotations

Adding annotations to a sequence can be done in two ways:

Open the sequence in a sequence view (double-click in the Navigation Area) | make a selection covering the part of the sequence you want to annotate¹ | 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 11.13.

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 11.3.1).
- **Type.** Reflects the left-hand part of the dialog as described above. You can also choose directly in this list or type your own annotation type.
- **Region.** If you have already made a selection, this field will show the positions of the selection. You can modify the region further using the conventions of DDBJ, EMBL

¹(See section 2.2.3 on how to make selections that are not contiguous.)

²Note 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, your own annotation type will be preserved

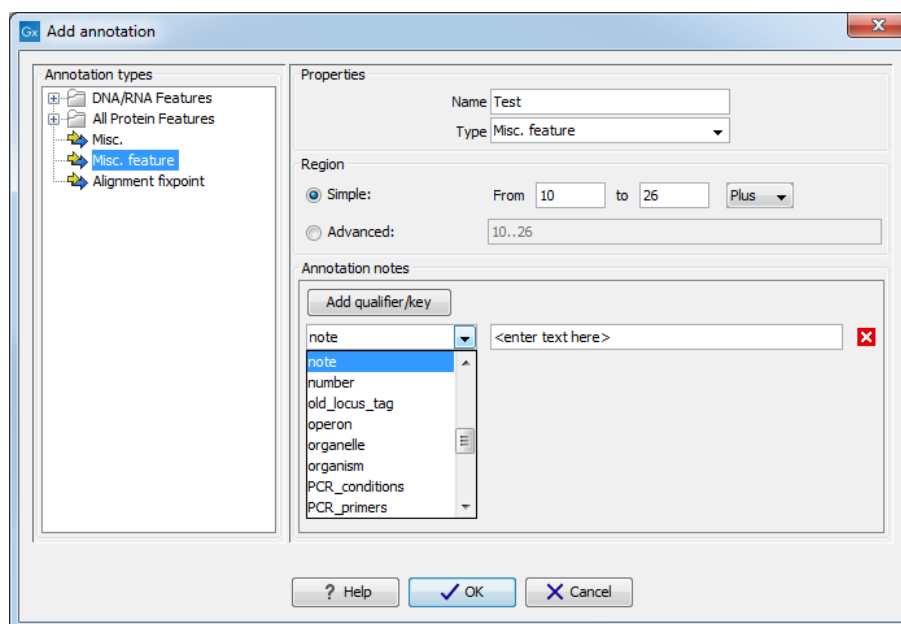


Figure 11.13: *The Add Annotation dialog.*

and GenBank. The following are examples of how to use the syntax (based on <http://www.ncbi.nlm.nih.gov/collab/FT/>):

- **467**. Points to a single residue in the presented sequence.
- **340..565**. Points to a continuous range of residues bounded by and including the starting and ending residues.
- **<345..500**. Indicates that the exact lower boundary point of a region is unknown. The location begins at some residue previous to the first residue specified (which is not 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.
- **123^124**. 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 11.3.1) and in the yellow box which appears when you place the mouse cursor on the annotation. If you write a hyperlink in the **Key** text field, like e.g. "www.qiagenbioinformatics.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.

11.3.3 Edit annotations

To edit an existing annotation from within a sequence view:

right-click the annotation | Edit Annotation (✎)

This will show the same dialog as in figure 11.13, 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 11.14.

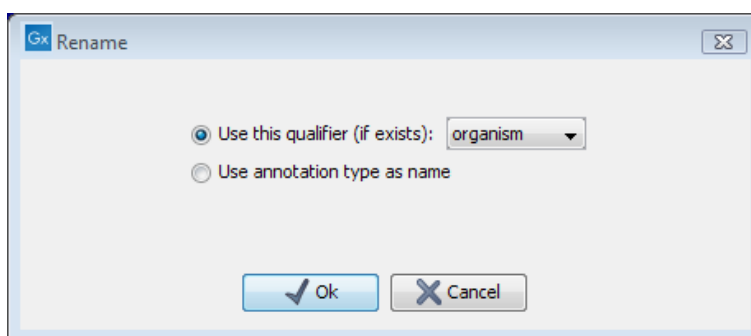


Figure 11.14: *The Advanced Rename dialog.*

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 11.15.

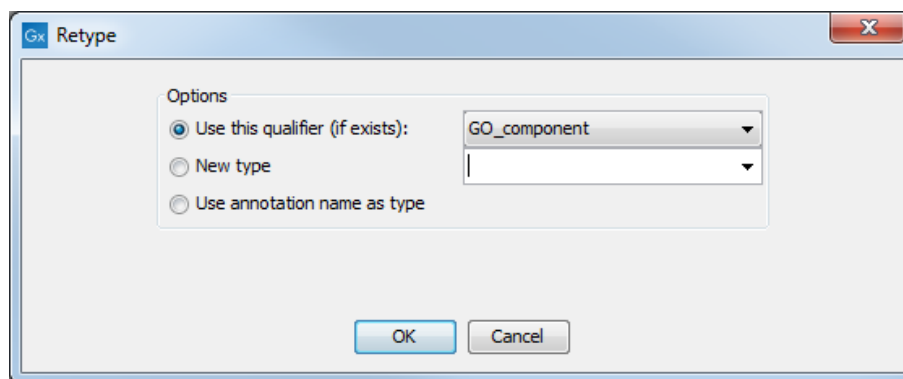


Figure 11.15: *The Advanced Retype dialog.*

In this dialog, you have two options:

- **Use this qualifier.** Use one of the qualifiers as type. A list of all qualifiers of all the selected annotations is shown. Note that if one of the annotations do not have the qualifier you have chosen, it will not be retyped. If an annotation has multiple qualifiers of the same type, the first is used for the new type.
- **New type.** You can select from a list of all the pre-defined types as well as enter your own annotation type. All the selected annotations will then get this type.
- **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).

11.3.4 Removing annotations

Annotations can be hidden using the **Annotation Types** preferences in the **Side Panel** to the right of the view (see section 11.3.1). 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

11.4 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 11.16.



Figure 11.16: 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. The information available depends on the origin of the sequence.

- **Name.** The name of the sequence which is also shown in sequence views and in the **Navigation Area**.
- **Description.** A description of the sequence.
- **Metadata.** The Metadata table and the detailed metadata values associated with 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 section 2.1.2 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.
- **Read group** Read group identifier "ID", technology used to produced the reads "Platform", and sample name "Sample".
- **Paired Status.** Unpaired or Paired sequences, with in this case the Minimum and Maximum distances as well as the Read orientation set during import.

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.

11.5 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 11.4.)

In the **Side Panel**, you find a search field for searching the text in the view.

11.6 Sequence Lists

The **Sequence List** is a file containing a number of sequences. Having sequences in a sequence list can help organizing sequence data. A Sequence List can be displayed in a graphical sequence view or in a tabular format. The two different views of the same sequence list are shown in split screen in figure 11.17.

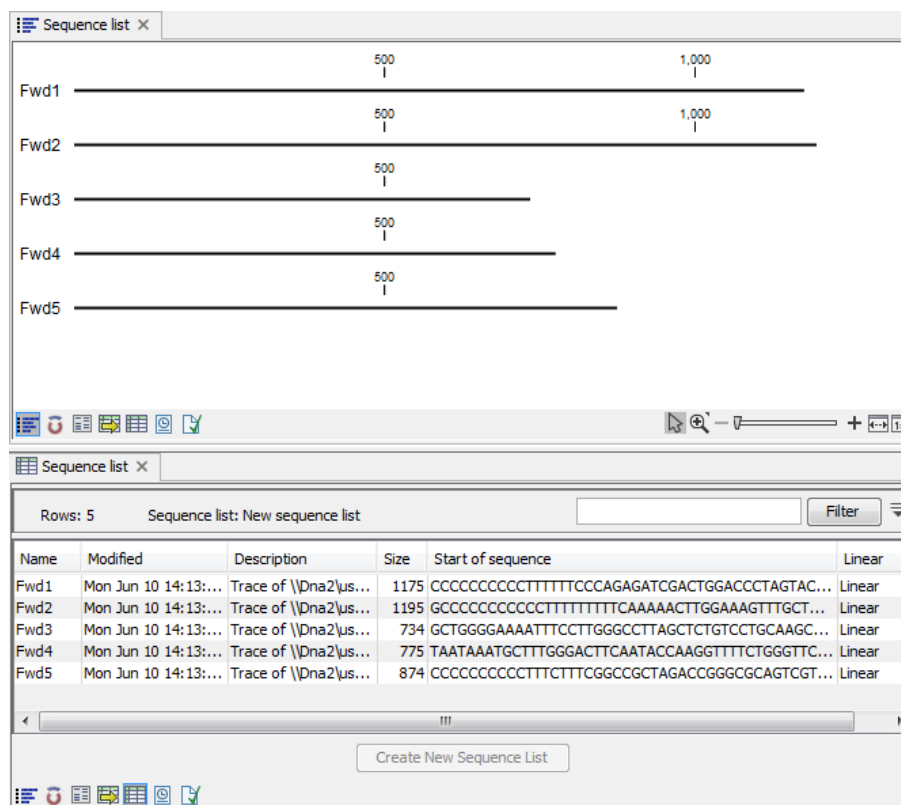


Figure 11.17: A sequence list containing multiple sequences can be viewed in either a table or in a graphical sequence list. The graphical view is useful for viewing annotations and the sequence itself, while the table view provides other information like sequence lengths, and the number of sequences in the list (number of Rows reported).

The **graphical view of sequence lists** is almost identical to the view of single sequences (see section 11.1). The main difference is that you now can see more than one sequence in the same view, and additionally 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**.

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** (⌘). To extract a sequence from a sequence list, drag the sequence directly from the table into the Navigation Area. Another option is to extract all sequences found in the list using the **Extract Sequences** tool. A description of how to use the **Extract Sequences** tool can be found in section 15.2.

Sequence lists are generated automatically when you import files containing more than one sequence. They may also be created as the output from particular Workbench tool, including database searches. For more information about creating sequence lists from a database search, see (chapter 7.1).

You can create a subset of a Sequence List: select the relevant sequences, right-click on the selected elements and choose **Create New Sequence List** from the drop down menu. This will generate a new sequence list that only includes the selected sequences.

A **Sequence List** can also be created from single sequences or by merging already existing sequence lists with the Workbench. To do this, select two or more sequences or sequence lists in the Navigation Area, right click on the selected elements and choose

New | Sequence List (⌘)

Alternatively, you can launch this tool via the "File" menu system.

This opens the **Sequence List** Wizard (figure 11.18). The dialog allows you to select more sequences to include in the list, or to remove already chosen sequences from the list.

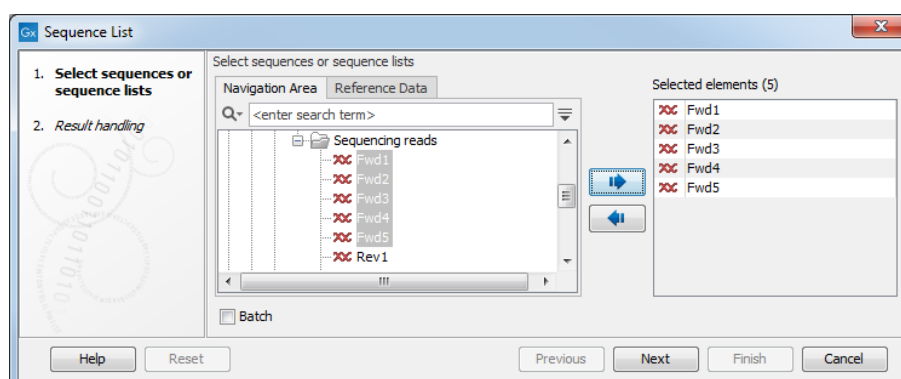


Figure 11.18: A Sequence List dialog.

If you are trying to create a new sequence list from a mixture of paired and unpaired datasets,

a warning message will let you know that the resulting sequence list will be set as unpaired (figure 11.19).

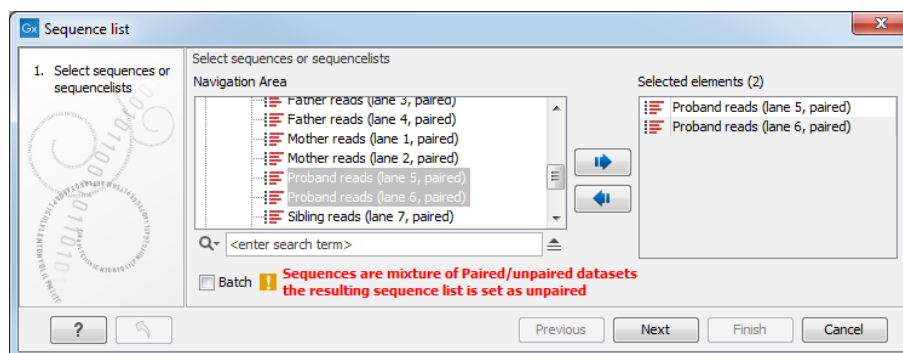


Figure 11.19: A warning appears when trying to create a new sequence list from a mixture of paired and unpaired datasets.

This warning also appears when trying to create a Sequence List out of paired reads lists for which the the Minimum and Maximum distances are different between lists. If that is the case, distances can be edited to be similar for all lists that needs to be merged in a new one.

For this, open all Sequence Lists one after the other and click on the Show Element Info icon at the bottom of the view (figure 11.20). Edit the distances by clicking on the button "Edit" next to the entry "Paired status" and click OK. Save the Sequence lists with the edited Paired statuses before attempting to create a merged sequence List. This final list's status will be set as Paired reads.

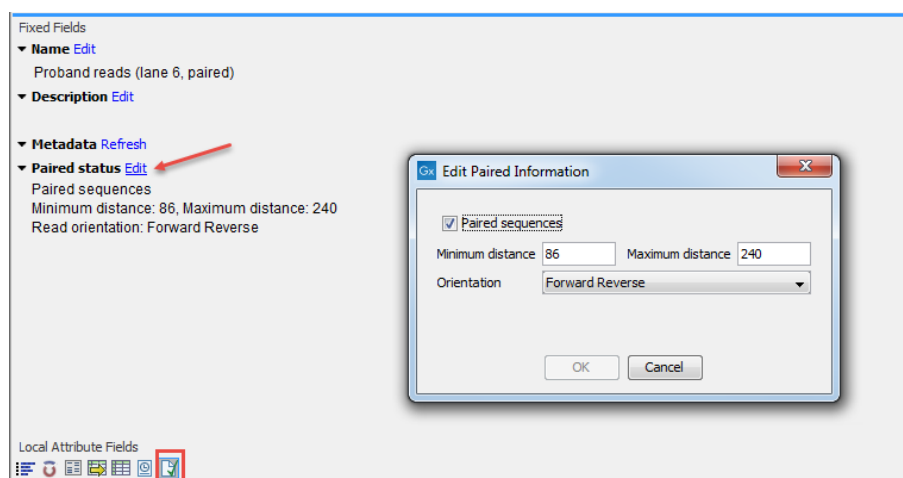


Figure 11.20: Edit the Minimum and Maximum distances of several sequence lists to be able to merge them into one.

Chapter 12

3D Molecule Viewer

Contents

12.1 Importing molecule structure files	219
12.1.1 From the Protein Data Bank	219
12.1.2 From your own file system	219
12.1.3 BLAST search against the PDB database	220
12.1.4 Import issues	221
12.2 Viewing molecular structures in 3D	222
12.2.1 Updating old structure files	223
12.3 Customizing the visualization	224
12.3.1 Visualization styles and colors	224
12.3.2 Project settings	230
12.4 Tools for linking sequence and structure	233
12.4.1 Show sequence associated with molecule	233
12.4.2 Link sequence or sequence alignment to structure	234
12.4.3 Transfer annotations between sequence and structure	235
12.5 Protein structure alignment	236
12.5.1 The Align Protein Structure dialog box	237
12.5.2 Example: alignment of calmodulin	237
12.5.3 The Align Protein Structure algorithm	239

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 molecular structures:

- 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 selected atoms
- Alignment of protein structures
- Browse amino acids and nucleic acids from sequence editors started from within the 3D Molecule Viewer
- Link a sequence or alignment to a protein structure
- Transfer annotations between the linked sequence and the structure

- 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 selected atoms
- Browse amino acids and nucleic acids from sequence editors started from within the 3D Molecule Viewer

12.1 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
- from your own file system
- using BLAST search against the PDB database

12.1.1 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 12.1). The search hits will appear in the table below the search field.

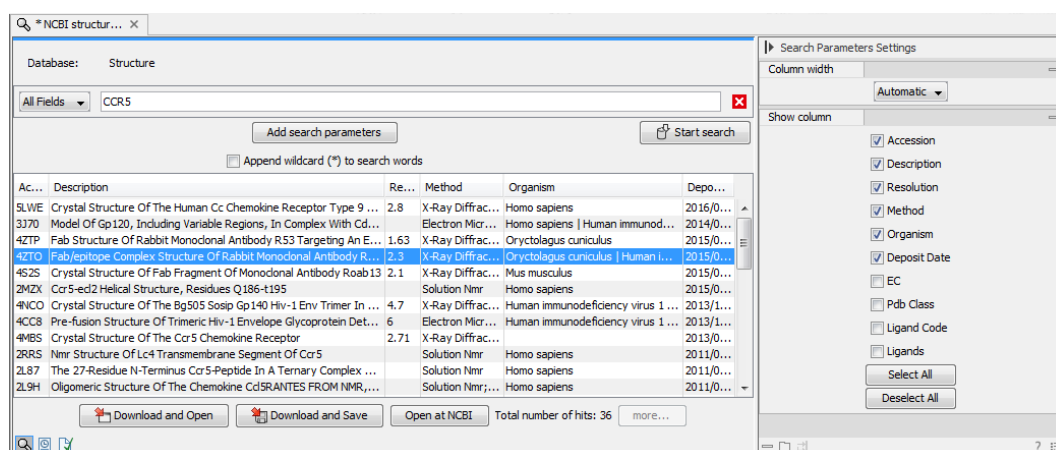


Figure 12.1: Download protein structure from the Protein Data Bank. It is possible to open a structure file directly from the output of the search by clicking the "Download and Open" button or by double clicking directly on the relevant row.

Select the molecule structure of interest and click on the button labeled "Download and Open" - 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.

12.1.2 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 12.2). 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**.

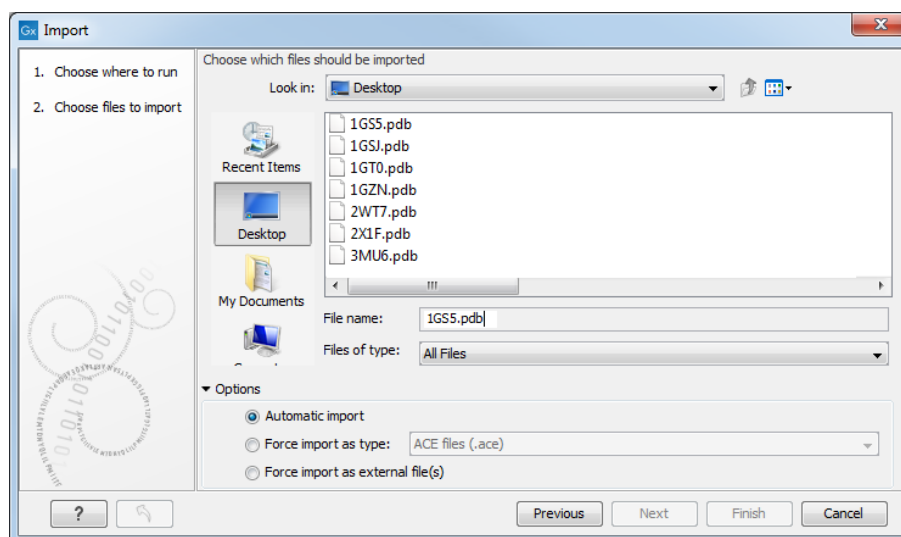


Figure 12.2: A PDB file can be imported using the Standard Import tool.

12.1.3 BLAST search against the PDB database

It is also possible to make a BLAST search against the PDB database, by going to:

Toolbox | BLAST (📄) | **BLAST at NCBI** (🌐)

After selecting where to run the analysis, specify which input sequences to use for the BLAST search in the "BLAST at NCBI" dialog, within the box named "Select sequences of same type". More than one sequence can be selected at the same time, as long as the sequences are of the same type (figure 12.3).

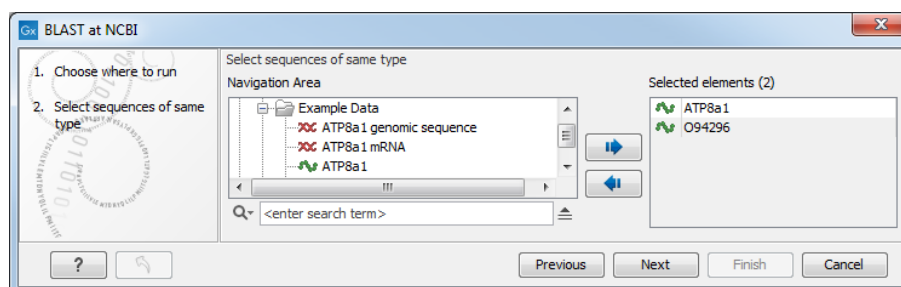


Figure 12.3: Select the input sequence of interest. In this example a protein sequence for ATPase class I type 8A member 1 and an ATPase ortholog from *S. pombe* have been selected.

Click **Next** and choose program and database (figure 12.4). 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.

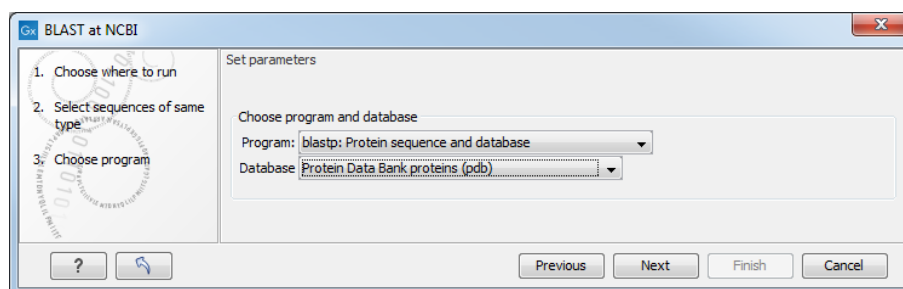


Figure 12.4: Select database and program.

Please refer to section 23.1.1 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 12.5). 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 selected structure in a **Molecule Project** in the **View Area**.

12.1.4 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 view. The Issues list can be opened by selecting **Show | Issues** from the menu appearing when right-clicking in an empty space in the 3D view (figure 12.6).

Alternatively, the issues can be accessed from the lower left corner of the view, where buttons are shown for each available view. If you hold down the Ctrl key (Cmd on Mac) while clicking on the Issues icon (🔍), the list will be shown in a split view together with the 3D view. The issues

The screenshot shows the BLAST interface with two main panels. The top panel displays sequence alignment results for a query sequence (ATP8a1) against several database sequences. The bottom panel shows a table of hits with columns for Hit, Description, E-value, Score, and %Gaps. The table contains 5 rows of data. Below the table are buttons for 'Download and Open', 'Download and Save', 'Open at NCBI', and 'Open Structure'. On the right side, there are settings for 'BLAST Settings' and 'BLAST HSP Table Settings'.

Hit	Description	E-value	Score	%Gaps
3TLM_A	Chain A, Crystal Structure Of Endoplasmic Reticulum Ca2+-Atpase (Serca) F...	1.77E-8	143.00	20.32
4YCM_A	Chain A, Crystal Structure Of The Calcium Pump With Bound Marine Macrolid...	4.40E-8	139.00	22.59
1KJU_A	Chain A, Ca2+-Atpase In The E2 State >gi 23200158 pdb 1IWO A Chain A...	4.48E-8	139.00	22.59
2DQS_A	Chain A, Crystal Structure Of The Calcium Pump With Amppcp In The Absen...	4.48E-8	139.00	22.59
3W5B_A	Chain A, Crystal Structure Of The Recombinant Serca1a (calcium Pump Of F...	4.48E-8	139.00	22.59
4NAB_A	Chain A, Structure Of The (sr)ca2+-atpase Mutant E309q In The Ca2-e1-mg...	5.09E-8	139.00	22.59
4BEW_A	Chain A, Serca Bound To Phosphate Analogue >gi 586500078 pdb 4BEW B	5.17E-8	139.00	22.59

Figure 12.5: 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.

list is linked with the molecules in the 3D view, such that selecting an entry in the list will select the implicated atoms in the view, and zoom to put them into the center of the 3D view.

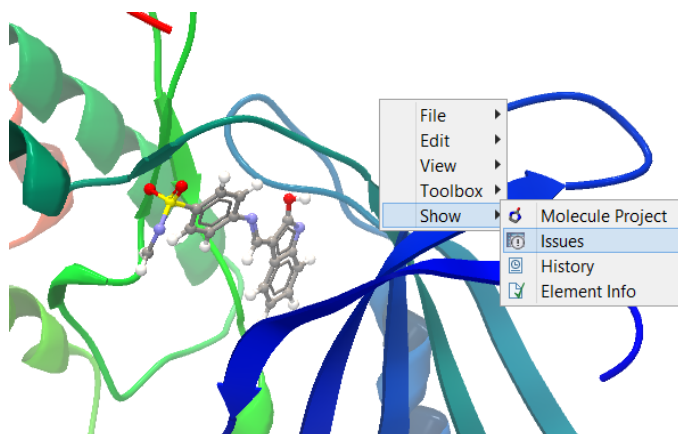


Figure 12.6: At the bottom of the Molecule Project it is possible to switch to the "Show Issues" view by clicking on the "table-with-exclamation-mark" icon.

12.2 Viewing molecular structures in 3D

An example of a 3D structure that has been opened as a **Molecule Project** is shown in figure 12.7.

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.

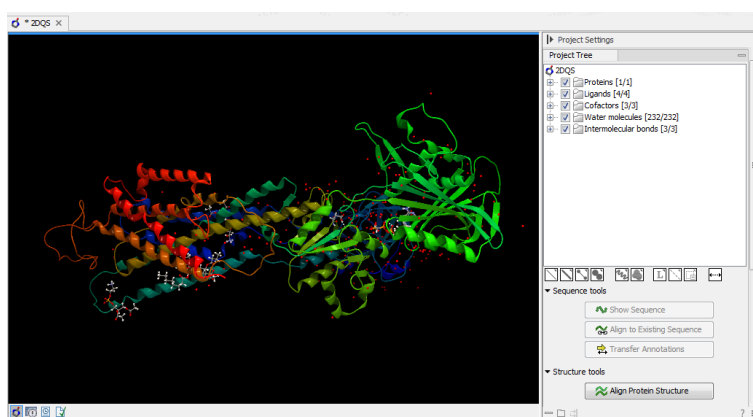


Figure 12.7: 3D view of a calcium ATPase. All molecules in the PDB file are shown in the Molecule Project. The Project Tree in the right side of the window lists the involved molecules.

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-checking the boxes next to them.

It is possible to bring a particular molecule or a category of molecules into focus by selecting 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 (\leftrightarrow) at the bottom of the **Project Tree** view.

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.

12.2.1 Updating old structure files

A completely redesign of the 3D Molecule Viewer was released in August 2013. It is therefore necessary to update older 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 12.8, which via the "Download from PDB..." button gives access to downloading the specific structure in PDB format.

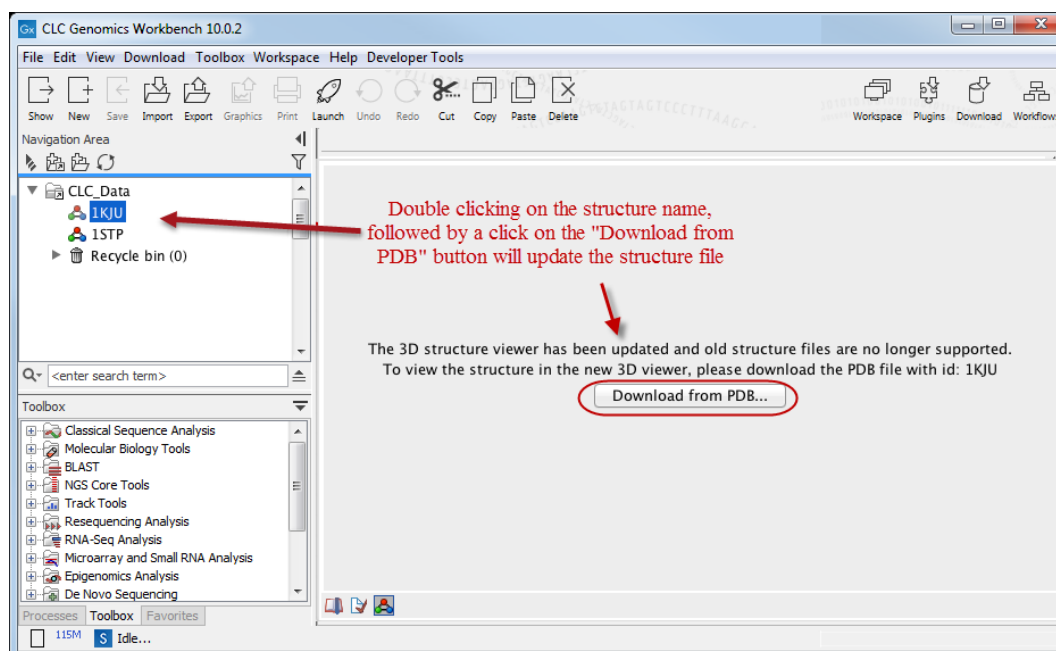


Figure 12.8: Old structure files are not supported by the new 3D Molecule Viewer and must be updated.

12.3 Customizing the visualization

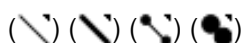
The molecular visualization of all molecules in the Molecule Project can be customized using different visualization styles. The styles can be applied to one molecule at a time, or to a whole category (or a mixture), by selecting the name of either the molecule or the category. Holding down the Ctrl (Cmd on Mac) or shift key while clicking the entry names in the **Project Tree** will select multiple molecules/categories.

The six leftmost 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**. Other quick-style buttons are available for displaying hydrogen bonds between Project Tree entries, for displaying labels in the 3D view and for creating custom atom groups. They are all described in detail below.

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.

12.3.1 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, and Space-filling/CPK.

The visualizations are mutually exclusive meaning that only one style can be applied at a time for

each selected molecule or atom group.

Six color schemes are available and can be accessed via right-clicking on the quick-style buttons:

- 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. For PDB files, this is based on the b-factors. For structure models created with tools in a CLC workbench, this is based on an estimate of the local model quality. The color scale goes from blue (0) over white (50) to red (100). The b-factors as well as the local model quality estimate are measures of uncertainty or disorder in the atom position; the higher the number, the higher the uncertainty.
- Color Carbons by Entry. Each entry (molecule or atom group) is assigned its own specific color. Only carbon atoms are colored by the specific color, other atoms are colored by element.
- Color by Entry. Each entry (molecule or atom group) is assigned its own specific color.
- Custom Color. The user selects a molecule color from a palette.
- Custom Carbon Color. The user selects a molecule color from a palette. Only carbon atoms are colored by the specific color, other atoms are colored by element.

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 Residue Position. Rainbow color scale going from blue over green to yellow and red, following the residue number.
- 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 Backbone Temperature. For PDB files, this is based on the b-factors for the C_{α} atoms (the central carbon atom in each amino acid). For structure models created with tools in the workbench, this is based on an estimate of the local model quality. The color scale goes from blue (0) over white (50) to red (100). The b-factors as well as the local model quality estimate are measures of uncertainty or disorder in the atom position; the higher the number, the higher the uncertainty.
- Color by Entry. Each chain/molecule is assigned its own specific color.
- Custom Color. The user selects a molecule color 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 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 temperature values assigned to atoms close to the surface (See the "Wireframe, Stick, Ball and stick, Space-filling/CPK" section above).
- Color by Entry. Each surface is assigned its own specific color.
- Custom Color. The user selects a surface color from a palette.

A surface spanning multiple molecules can be visualized by creating a custom atom group that includes all atoms from the molecules (see section 12.3.1)

It is possible to adjust the opacity of a surface by adjusting the transparency slider at the bottom of the menu.

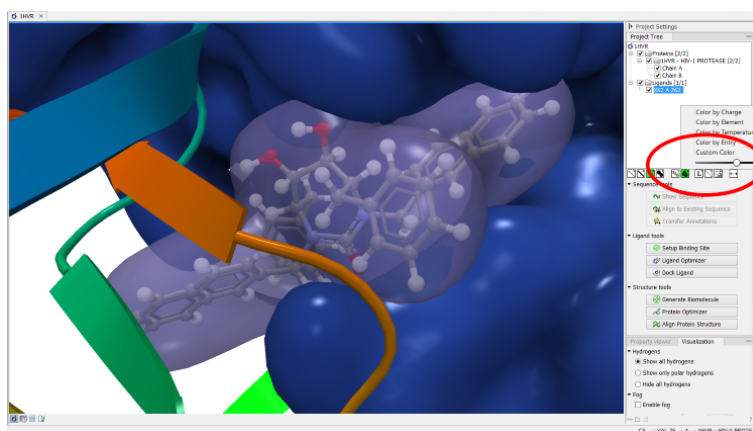


Figure 12.9: Transparent surfaces

Notice that visual artifacts may appear when rotating a transparent surface. These artifacts disappear as soon as the mouse is released.

Labels



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 12.10).

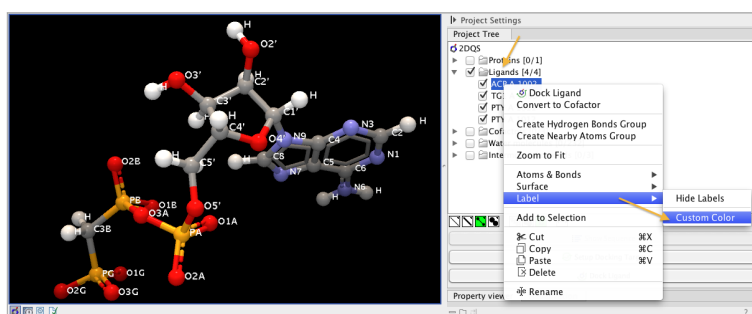


Figure 12.10: The color of the labels can be adjusted in two different ways. Either directly using the label button by right clicking the button, or by right clicking on the molecule or category of interest in the Project Tree.

- For proteins and nucleic acids, each residue is labeled with the PDB name and number.
- For ligands, each atom is labeled with the atom name as given in the input.
- For cofactors and water, one label is added with the name of the molecule.
- For atom groups including protein atoms, each protein residue is labeled with the PDB name and number.
- For atom groups not including protein atoms, each atom is labeled with the atom name as given in the input.

Labels can be removed again by clicking on the label button.

Hydrogen bonds



The Show Hydrogen Bond visualization style may be applied to molecules and atom group entries in the project tree. If this style is enabled for a project tree entry, hydrogen bonds will be shown to all other currently visible objects. The hydrogen bonds are updated dynamically: if a molecule is toggled off, the hydrogen bonds to it will not be shown.

It is possible to customize the color of the hydrogen bonds using the context menu.

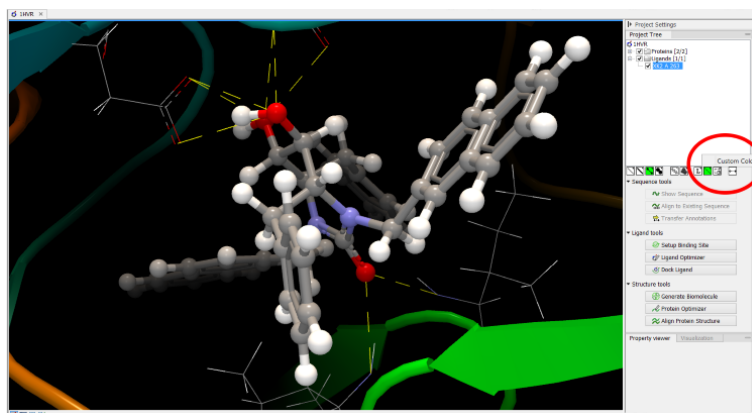


Figure 12.11: The hydrogen bond visualization setting, with custom bond color.

Create atom group



Often it is convenient 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 achieved by creating an atom group. Atom groups can be created based on atoms selected in the 3D view or entries selected in the Project Tree. When an atom group has been created, it appears as an entry in the Project Tree in the category "Atom groups". The atoms can then be hidden or shown, and the visualization changed, just as for the molecule entries in the Project Tree.

Note that an atom group entry can be renamed. Select the atom group in the Project Tree and invoke the right-click context menu. Here, the Rename option is found.

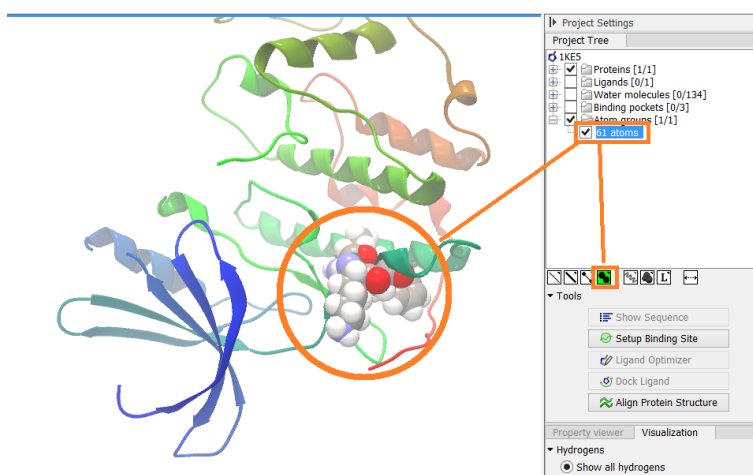


Figure 12.12: An atom group that has been highlighted by adding a unique visualization style.

Create atom group based on atoms selected in 3D view

When atoms are selected in the 3D view, brown spheres indicate which atoms are included in the selection. The selection will appear as the entry "Current" in the Selections category in the Project Tree.

Once a selection has been made, press the "Create Atom Group" button and a context menu will show different options for creating a new atom group based on the selection:

- **Selected Atoms.** Creates an atom group containing exactly the selected atoms (those indicated by brown spheres). If an entire molecule or residue is selected, this option is not displayed.
- **Selected Residue(s)/Molecules.** Creates an atom group that includes all atoms in the selected residues (for entries in the protein and nucleic acid categories) and molecules (for the other categories).
- **Nearby Atoms.** Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) within 5 Å of the selected atoms. Only atoms from currently visible Project Tree entries are considered.
- **Hydrogen Bonded Atoms.** Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) that have hydrogen

bonds to the selected atoms. Only atoms from currently visible Project Tree entries are considered.

There are several ways to select atoms in the 3D view:

- 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 to the "Current" selection by choosing "Add to Current Selection" in the context menu. Similarly, entire 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 atoms inside the sphere, visualized with one of the all-atom representations 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 (section 12.4.1). A split-view will appear with a sequence editor for each of the sequence data types (Protein, DNA, RNA) (figure 12.13). If you then select residues in the sequence view, the backbone atoms of 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 sequence is modified.
- Align to Existing Sequence. If a single protein chain is selected in the Project Tree, the "Align to Existing Sequence" button can be clicked (section 12.4.2). This links the protein sequence with a sequence or sequence alignment found in the Navigation Area. A split-view appears with a sequence alignment where the sequence of the selected protein chain is linked to the 3D structure, and atoms can be selected in the 3D view, just as for the "Show Sequence" option.

Create atom group based on entries selected in the Project Tree

Select one or more entries in the Project Tree, and press the "Create Atom Group" button, then a context menu will show different options for creating a new atom group based on the selected entries:

- Nearby Atoms. Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) within 5 Å of the selected entries. Only atoms from currently visible Project Tree entries are considered.
- Hydrogen Bonded Atoms. Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) that have hydrogen bonds to the selected entries. Only atoms from currently visible Project Tree entries are considered.

If a Binding Site Setup is present in the Project Tree (A Binding Site Setup could only be created using the now discontinued CLC Drug Discovery Workbench), and entries from the Ligands or

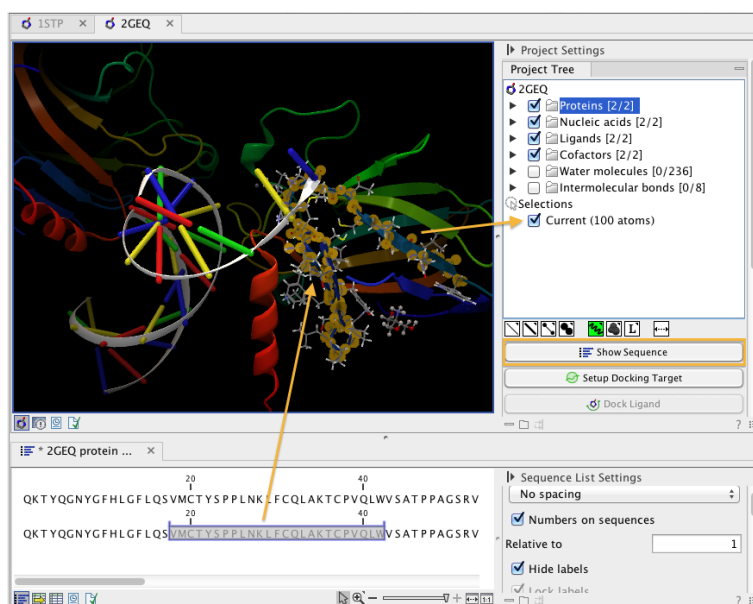


Figure 12.13: The protein sequence in the split view is linked with the protein structure. This means that when a part of the protein sequence is selected, the same region in the protein structure will be selected.

Docking results categories are selected, two extra options are available under the header **Create Atom Group (Binding Site)**. For these options, atom groups are created considering all molecules included in the Binding Site Setup, and thus not taking into account which Project Tree entries are currently visible.

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 12.14). Double-clicking an entry in the Project Tree will have the same effect.

12.3.2 Project settings

A number of general settings can be adjusted from the **Side Panel**. Personal settings as well as molecule visualizations can be saved by clicking in the lower right corner of the **Side Panel** (☰). This is described in detail in section 4.6.

Project Tree Tools

Just below the Project Tree, the following tools are 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

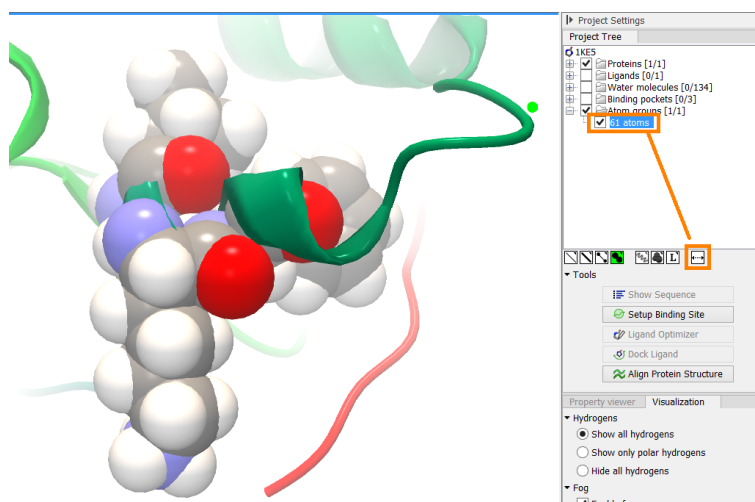


Figure 12.14: The "Fit to screen" button can be used to bring a particular molecule or category of molecules in focus.

editor for each of the sequence data types (Protein, DNA, RNA). This is described in section [12.4.1](#).

- **Align to Existing Sequence** Select a protein chain in the Project Tree, and click this button. Then protein sequences and sequence alignments found in the Navigation Area, can be linked with the protein structure. This is described in section [12.4.2](#).
- **Transfer Annotations** Select a protein chain in the Project Tree, that has been linked with a sequence using either the "Show Sequence" or "Align to Existing Sequence" options. Then it is possible to transfer annotations between the structure and the linked sequence. This is described in section [12.4.3](#).
- **Align Protein Structure** This will invoke the dialog for aligning protein structures based on global alignment of whole chains or local alignment of e.g. binding sites defined by atom groups. This is described in section [12.5](#).

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:

- **Molecule** The name of the molecule the atom is part of.
- **Residue** For proteins and nucleic acids, the name and number of the residue the atom belongs to is listed, and the chain name is displayed in parentheses.
- **Name** The particular atom name, if given in input, with the element type (Carbon, Nitrogen, Oxygen...) displayed in parentheses.
- **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.

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.
- **Occupancy** For each atom in a PDB file, the occupancy is given. It is typically 1, but if atoms are modeled in the PDB file, with no foundation in the raw data, the occupancy is 0. If a residue or molecule has been resolved in multiple positions, the occupancy is between 0 and 1.

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.

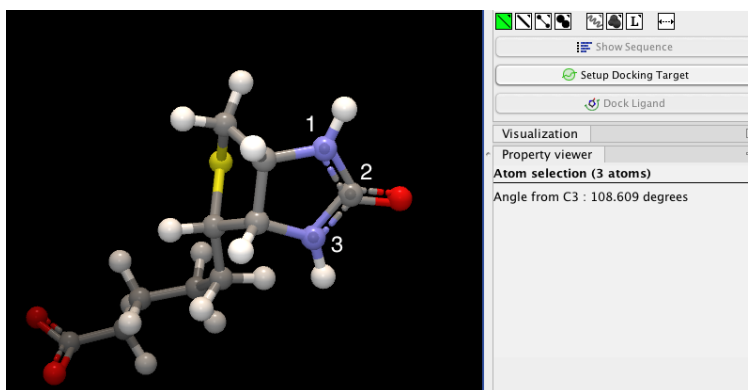


Figure 12.15: Selecting two, three, or four atoms will display the distance, angle, or dihedral angle, respectively.

If a molecule is selected in the Project Tree, the Property view shows information about this molecule. Two measures are always shown:

- **Atoms** Number of atoms in the molecule.
- **Weight** The weight of the molecule in Daltons.

Visualization settings

Under "Visualization" five 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.
- **Clipping plane** This option makes it possible to add an imaginary plane at a specified distance along the camera's line of sight. Only objects behind this plane will be drawn. It is possible to clip only surfaces, or to clip surfaces together with proteins and nucleic acids. Small molecules, like ligands and water molecules, are never clipped.

- **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.

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.

You can also save the current view directly on data with a custom name, so that it can later be applied (see section 4.6).

12.4 Tools for linking sequence and structure

The *CLC Main Workbench* has functionality that allows you to link a protein sequence to a protein structure. Selections made on the sequence will show up on the structure. This allows you to explore a protein sequence in a 3D structure context. Furthermore, sequence annotations can be transferred to annotations on the structure and annotations on the structure can be transferred to annotations on the sequence (see section 12.4.3).

12.4.1 Show sequence associated with molecule

From the Side Panel, sequences associated with the molecules in the Molecule Project can be opened as separate objects by selecting protein or nucleic acid entries in the Project Tree and clicking the button labeled "Show Sequence" (figure 12.16). This will generate a Sequence or Sequence List for each selected sequence type (protein, DNA, RNA). The sequences can be used to select atoms in the Molecular Project as described in section 12.3.1. The sequences can also be used as input for sequence analysis tools or be saved as independent objects. You can later re-link to the sequence using "Align to Existing Sequence" (see section 12.4.2).

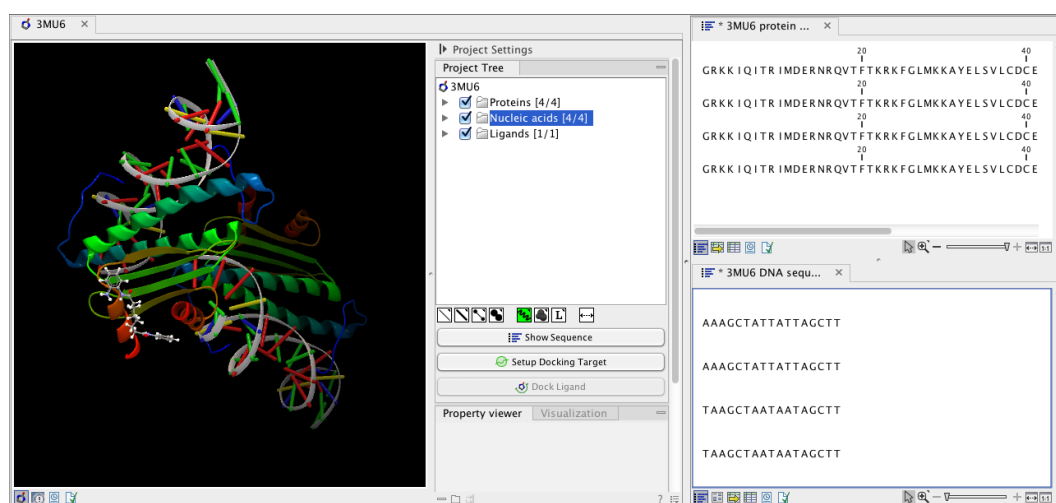


Figure 12.16: Protein chain sequences and DNA sequences are shown in separate views.

12.4.2 Link sequence or sequence alignment to structure

The "Align to Existing Sequence" button can be used to map and link existing sequences or sequence alignments to a protein structure chain in a Molecule Project (3D view). It can also be used to reconnect a protein structure chain to a sequence or sequence alignment previously created by Show Sequence (section 12.4.1) or Align to Existing Sequence.

Select a single protein chain in the project tree (see figure 12.17). Pressing "Align to Existing Sequence" then opens a Navigation Area browser, where it is possible to select one or more Sequence, Sequence Lists, or Alignments, to link with the selected protein chain.

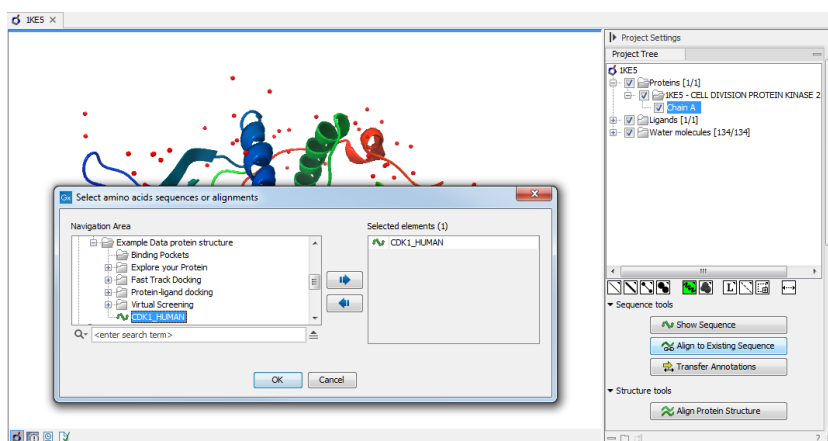


Figure 12.17: Select a single protein chain in the Project Tree and invoke "Align to Existing Sequence".

If the sequences or alignments already contain a sequence identical to the protein chain selected in the Molecule Project (i.e. same name and amino acid sequence), this sequence is linked to the protein structure. If no identical sequence is present, a sequence is extracted from the protein structure (as for Show Sequence - section 12.4.1), and a sequence alignment is created between this sequence and the sequences or alignments selected from the Navigation Area. The new sequence alignment is created (see section 13.1) with the following settings:

- Gap open cost: 10.0
- Gap Extension cost: 1.0
- End gap cost: free
- Existing alignments are not redone

When the link is established, selections on the linked sequence in the sequence editor will create atom selections in the 3D view, and it is possible to transfer annotations between the linked sequence and the 3D protein chain (see section 12.4.3). Notice, that the link will be broken if either the sequence or the 3D protein chain is modified.

Two tips if the link is to a sequence in an alignment:

1. Read about how to change the layout of sequence alignments in section 13.2
2. It is only annotations present on the sequence linked to the 3D view that can be transferred to atom groups on the structure. To transfer sequence annotations from other sequences

in the alignment, first copy the annotations to the sequence in the alignment that is linked to the structure (see figure 12.20 and section 13.3).

12.4.3 Transfer annotations between sequence and structure

The Transfer Annotations dialog makes it possible to create new atom groups (annotations on structure) based on protein sequence annotations and vice versa.

You can read more about sequence annotations in section 11.3 and more about atom groups in section 12.3.1.

Before it is possible to transfer annotations, a link between a protein sequence editor and a Molecule Project (a 3D view) must be established. This is done either by opening a sequence associated with a protein chain in the 3D view using the 'Show Sequence' button (see section 12.4.1) or by mapping to an existing sequence or sequence alignment using the 'Align to Existing Sequence' button (see section 12.4.2).

Invoke the Transfer Annotations dialog by selecting a linked protein chain in the Project Tree and press 'Transfer Annotations' (see figure 12.18).

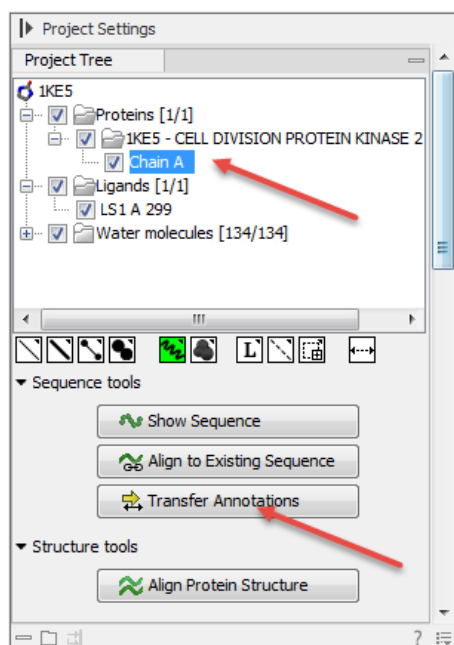


Figure 12.18: Select a single protein chain in the Project Tree and invoke "Transfer Annotations".

The dialog contains two tables (see figure 12.19). The left table shows all atom groups in the Molecule Project, with at least one atom on the selected protein chain. The right table shows all annotations present on the linked sequence. While the Transfer Annotations dialog is open, it is not possible to make changes to neither the sequence nor the Molecule Project, however, changes to the visualization styles are allowed.

How to undo annotation transfers

In order to undo operations made using the Transfer Annotations dialog, the dialog must first be closed. To undo atom groups added to the structure, activate the 3D view by clicking in it and press Undo in the Toolbar. To undo annotations added to the sequence, activate the sequence view by clicking in it and press Undo in the Toolbar.

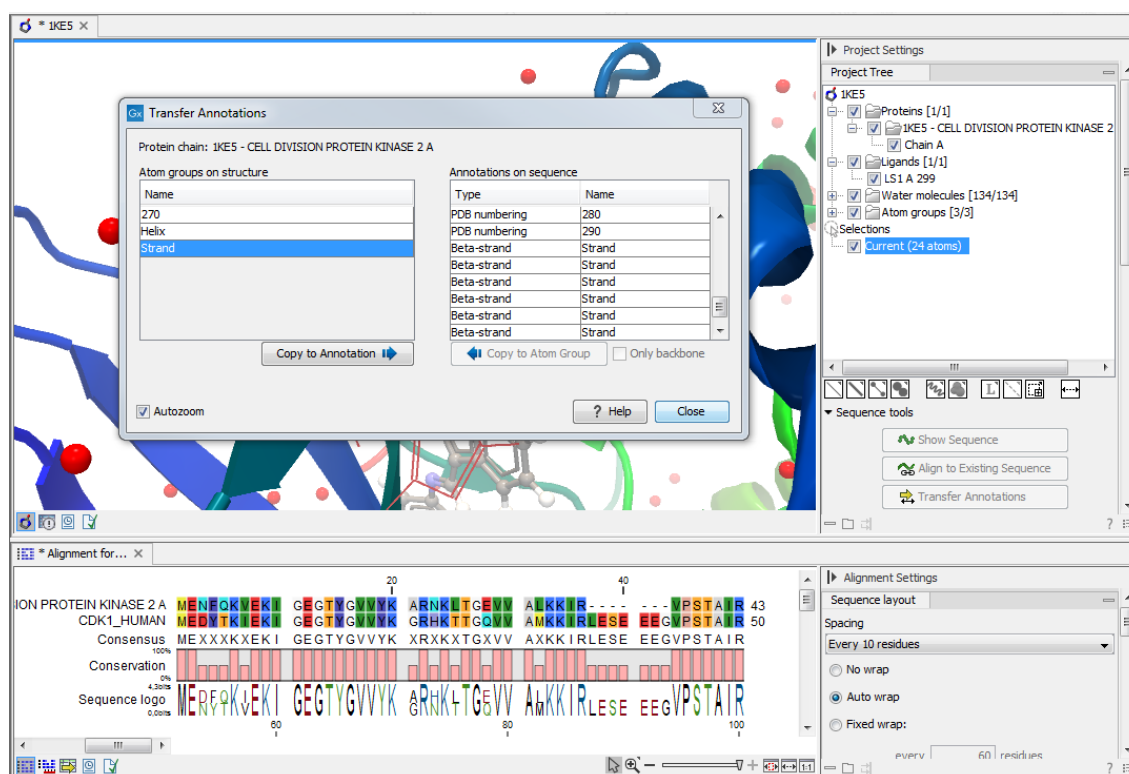


Figure 12.19: The Transfer Annotations dialog allow you to select annotations listed in the two tables, and copy them from structure to sequence or vice versa.

Transfer sequence annotations from aligned sequences

It is only annotations present on the sequence linked to the 3D view that can be transferred to atom groups on the structure. If you wish to transfer annotations that are found on other sequences in a linked sequence alignment, you need first to copy the sequence annotations to the actual sequence linked to the 3D view (the sequence with the same name as the protein structure). This is done by invoking the context menu on the sequence annotation you wish to copy (see figure 12.20 and section 13.3).

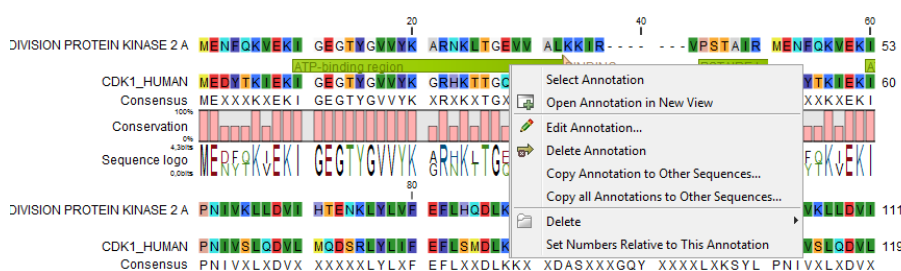


Figure 12.20: Copy annotations from sequences in the alignment to the sequence linked to the 3D view.

12.5 Protein structure alignment

The Align Protein Structure tool allows you to compare a protein or binding pocket in a **Molecule Project** with proteins from other **Molecule Projects**. The tool is invoked using the  Align Protein Structure action from the **Molecule Project Side Panel**. This action will open an interactive

dialog box (figure 12.21). By default, when the dialog box is closed with an "OK", a new **Molecule Project** will be opened containing all the input protein structures laid on top of one another. All molecules coming from the same input Molecule Project will have the same color in the initial visualization.

12.5.1 The Align Protein Structure dialog box

The dialog box contains three fields:

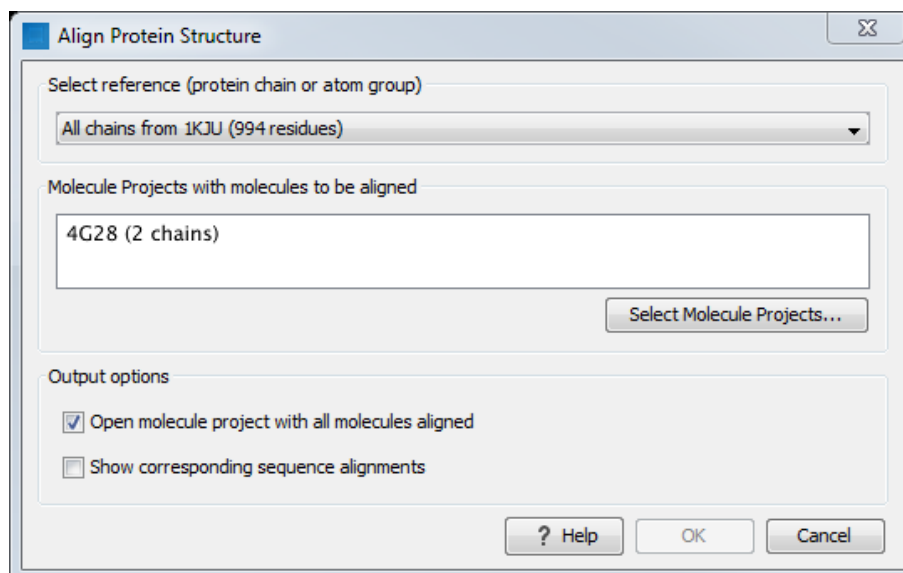


Figure 12.21: The Align Protein Structure dialog box.

- **Select reference (protein chain or atom group)** This drop-down menu shows all the protein chains and residue-containing atom groups in the current **Molecule Project**. If an atom group is selected, the structural alignment will be optimized in that area. The 'All chains from Molecule Project' option will create a global alignment to all protein chains in the project, fitting e.g. a dimer to a dimer.
- **Molecule Projects with molecules to be aligned** One or more **Molecule Projects** containing protein chains may be selected.
- **Output options** The default output is a single **Molecule Project** containing all the input projects rotated onto the coordinate system of the reference. Several alignment statistics, including the RMSD, TM-score, and sequence identity, are added to the **History** of the output **Molecule Project**. Additionally, a sequence alignments of the aligned structures may be output, with the sequences linked to the 3D structure view.

12.5.2 Example: alignment of calmodulin

Calmodulin is a calcium binding protein. It is composed of two similar domains, each of which binds two calcium atoms. The protein is especially flexible, which can make structure alignment challenging. Here we will compare the calcium binding loops of two calmodulin crystal structures – PDB codes 1A29 and 4G28.

Initial global alignment The 1A29 project is opened and the Align Protein Structure dialog is filled out as in figure 12.21. Selecting "All chains from 1A29" tells the aligner to make the best possible global alignment, favoring no particular region. The output of the alignment is shown in figure 12.22. The blue chain is from 1A29, the brown chain is the corresponding calmodulin chain from 4G28 (a calmodulin-binding chain from the 4G28 file has been hidden from the view). Because calmodulin is so flexible, it is not possible to align both of its domains (enclosed in black boxes) at the same time. A good global alignment would require the brown protein to be translated in one direction to match the N-terminal domain, and in the other direction to match the C-terminal domain (see black arrows).

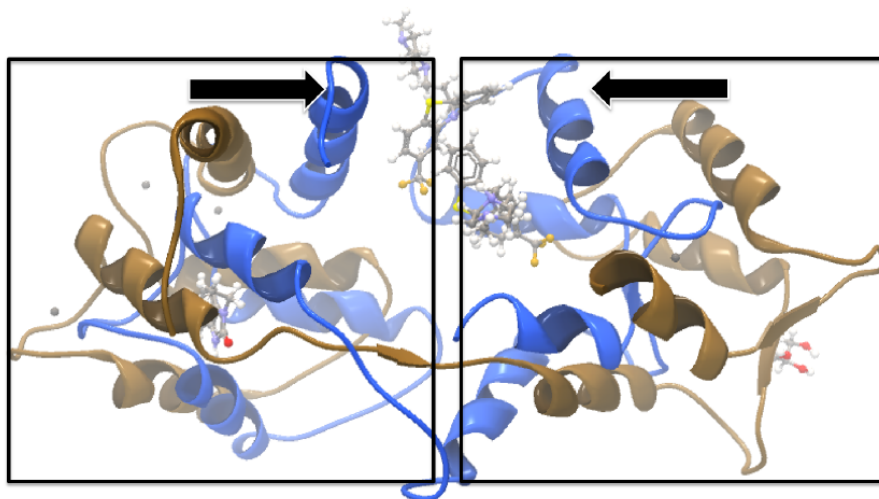


Figure 12.22: Global alignment of two calmodulin structures (blue and brown). The two domains of calmodulin (shown within black boxes) can undergo large changes in relative orientation. In this case, the different orientation of the domains in the blue and brown structures makes a good global alignment impossible: the movement required to align the brown structure onto the blue is shown by arrows – as the arrows point in opposite directions, improving the alignment of one domain comes at the cost of worsening the alignment of the other.

Focusing the alignment on the N-terminal domain To align only the N-terminal domain, we return to the 1A29 project and select the **Show Sequence** action from beneath the **Project Tree**. We highlight the first 62 residues, then convert them into an atom group by right-clicking on the "Current" selection in the **Project Tree** and choosing "Create Group from Selection" (figure 12.23). Using the new atom group as the reference in the alignment dialog leads to the alignment shown in figure 12.24. In addition to the original input proteins, the output now includes two Atom Groups, which contain the atoms on which the alignment was focused. The **History** of the output **Molecule Project** shows that the alignment has 0.9 Å RMSD over the 62 residues.

Aligning a binding site Two bound calcium atoms, one from each calmodulin structure, are shown in the black box of figure 12.24. We now wish to make an alignment that is as good as possible about these atoms so as to compare the binding modes. We return to the 1A29 project, right-click the calcium atom from the cofactors list in the **Project Tree** and select "Create Nearby Atoms Group". Using the new atom group as the reference in the alignment dialog leads to the alignment shown in figure 12.25.

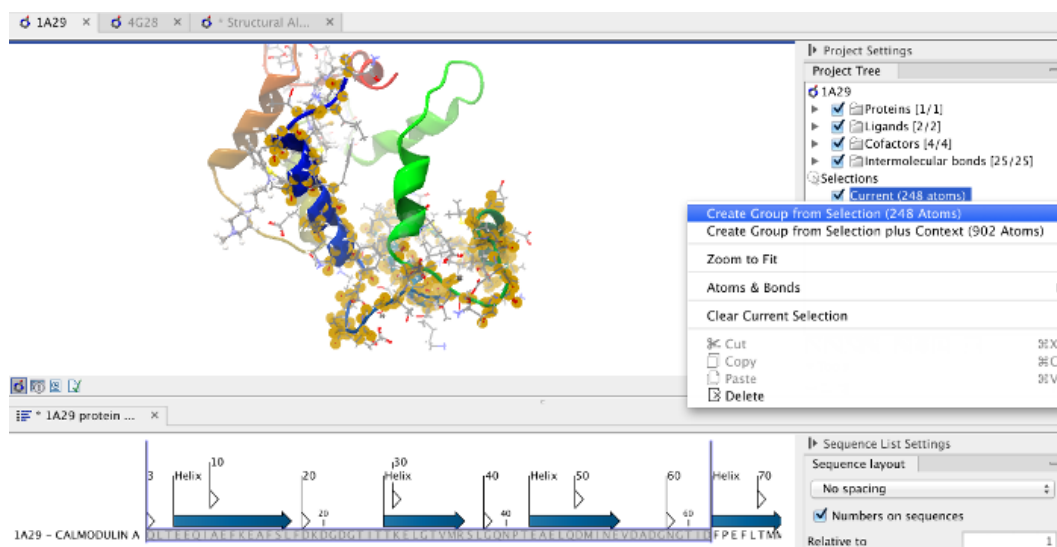


Figure 12.23: Creation of an atom group containing the N-terminal domain of calmodulin.

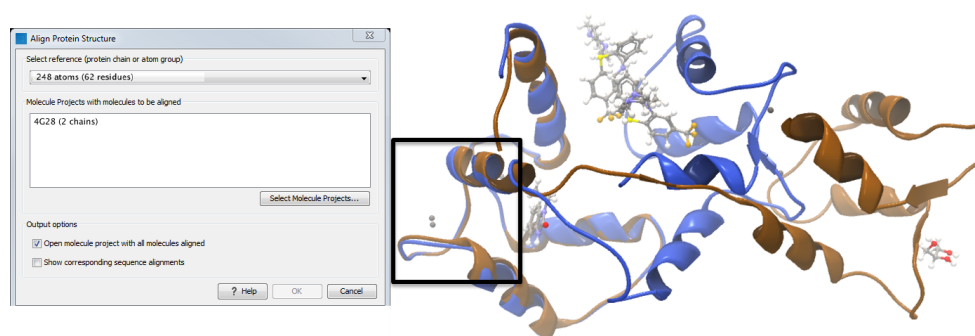


Figure 12.24: Alignment of the same two calmodulin proteins as in figure 12.22, but this time with a focus on the N-terminal domain. The blue and brown structures are now well-superimposed in the N-terminal region. The black box encloses two calcium atoms that are bound to the structures.

12.5.3 The Align Protein Structure algorithm

Any approach to structure alignment must make a trade-off between alignment length and alignment accuracy. For example, is it better to align 200 amino acids at an RMSD of 3.0 Å or 150 amino acids at an RMSD of 2.5 Å? The Align Protein Structure algorithm determines the answer to this question by taking the alignment with the higher TM-score. For an alignment focused on a protein of length L , this is:

$$\text{TM-score} = \frac{1}{L} \sum_i \frac{1}{1 + \frac{d_i}{d(L)^2}}$$

where i runs over the aligned pairs of residues, d_i is the distance between the i^{th} such pair, and $d(L)$ is a normalization term that approximates the average distance between two randomly chosen points in a globular protein of length L [Zhang and Skolnick, 2004]. A perfect alignment has a TM-score of 1.0, and two proteins with a TM-score >0.5 are often said to show structural homology [Xu and Zhang, 2010].

The Align Protein Structure Algorithm attempts to find the *structure alignment* with the highest

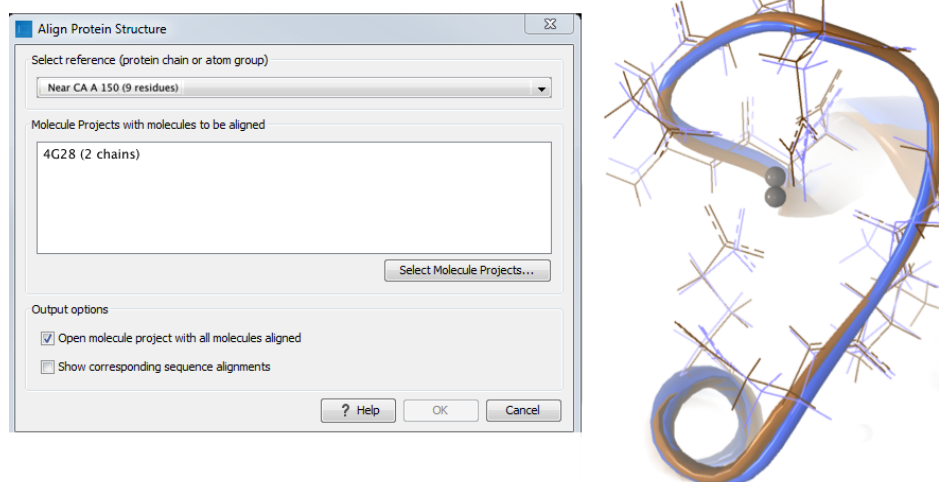


Figure 12.25: Alignment of the same two calmodulin domains as in figure 12.22, but this time with a focus on the calcium atom within the black box of figure 12.24. The calcium atoms are less than 1 Å apart – compatible with thermal motion encoded in the atoms' temperature factors.

TM-score. This problem reduces to finding a *sequence alignment* that pairs residues in a way that results in a high TM-score. Several sequence alignments are tried including an alignment with the BLOSUM62 matrix, an alignment of secondary structure elements, and iterative refinements of these alignments.

The Align Protein Structure Algorithm is also capable of aligning entire protein complexes. To do this, it must determine the correct pairing of each chain in one complex with a chain in the other. This set of chain pairings is determined by the following procedure:

1. Make structure alignments between every chain in one complex and every chain in the other. Discard pairs of chains that have a TM-score of < 0.4
2. Find all pairs of structure alignments that are consistent with each other i.e. are achieved by approximately the same rotation
3. Use a heuristic to combine consistent pairs of structure alignments into a single alignment

The heuristic used in the last step is similar to that of MM-align [Mukherjee and Zhang, 2009], whereas the first two steps lead to both a considerable speed up and increased accuracy. The alignment of two 30S ribosome subunits, each with 20 protein chains, can be achieved in less than a minute (PDB codes 2QBD and 1FJG).

Chapter 13

Sequence alignment

Contents

13.1 Create an alignment	241
13.1.1 Gap costs	242
13.1.2 Fast or accurate alignment algorithm	243
13.1.3 Aligning alignments	244
13.1.4 Fixpoints	244
13.2 View alignments	246
13.2.1 Bioinformatics explained: Sequence logo	249
13.3 Edit alignments	250
13.3.1 Realignment	252
13.4 Join alignments	254
13.5 Pairwise comparison	255
13.5.1 The pairwise comparison table	257
13.5.2 Bioinformatics explained: Multiple alignments	258

CLC Main Workbench can align nucleotides and proteins using a *progressive alignment* algorithm (see section 13.5.2).

This chapter describes how to use the program to align sequences, and alignment algorithms in more general terms.

13.1 Create an alignment

Alignments can be created from sequences, sequence lists (see section 11.6), existing alignments and from any combination of the three.

To create an alignment in *CLC Main Workbench*:

Toolbox | Alignments and Trees (📁) | Create Alignment (🔍)

This opens the dialog shown in figure 13.1.

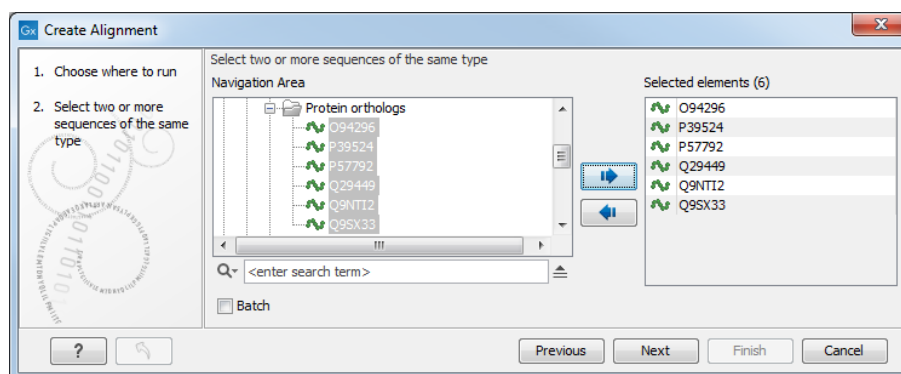


Figure 13.1: Creating an alignment.

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 13.2.

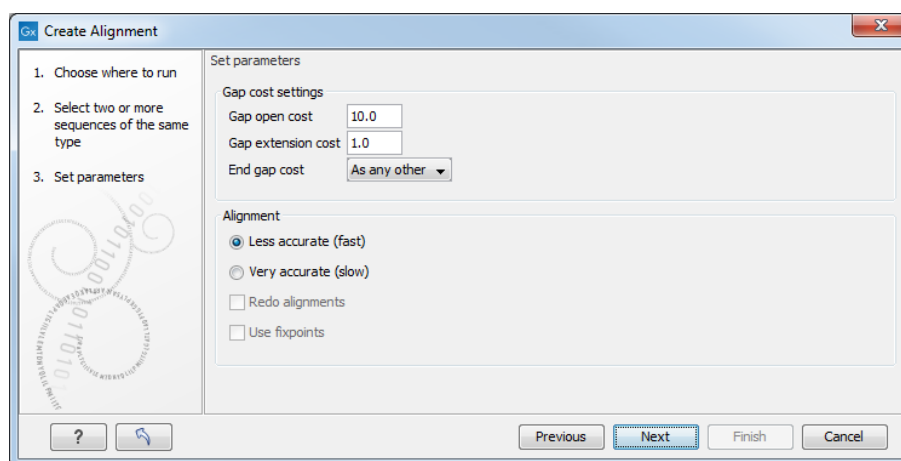


Figure 13.2: Adjusting alignment algorithm parameters.

13.1.1 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 13.3 and 13.4 illustrate the differences between the different gap scores at the sequence ends.

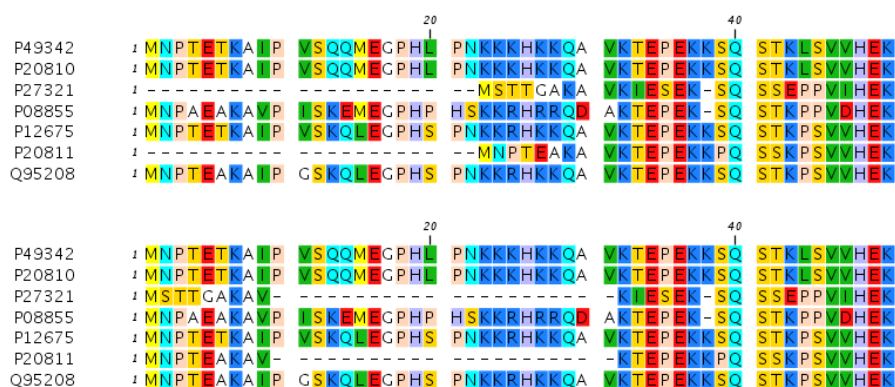


Figure 13.3: The first 50 positions of two different alignments of seven calpastatin sequences. The top alignment is made with cheap end gaps, while the bottom alignment is made with end gaps having the same price as any other gaps. In this case it seems that the latter scoring scheme gives the best result.

13.1.2 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.

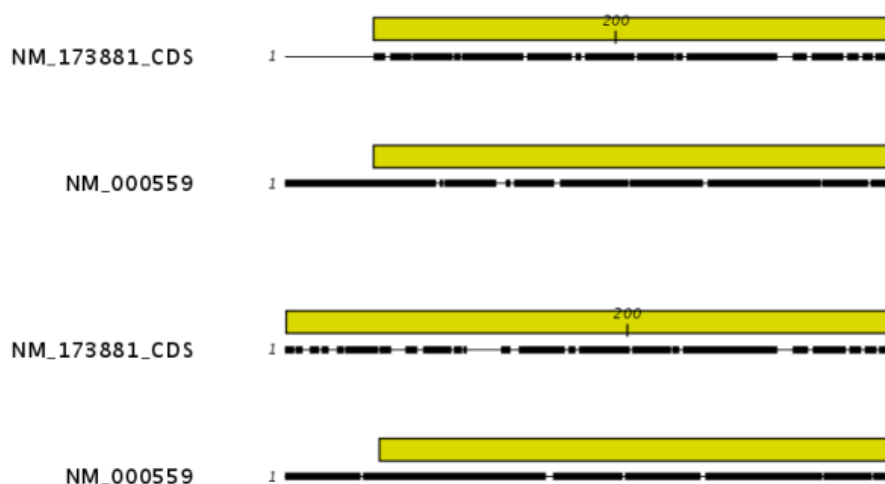


Figure 13.4: The alignment of the coding sequence of bovine myoglobin with the full mRNA of human gamma globin. The top alignment is made with free end gaps, while the bottom alignment is made with end gaps treated as any other. The yellow annotation is the coding sequence in both sequences. It is evident 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.

13.1.3 Aligning alignments

If you have selected an existing alignment in the first step (13.1), 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 13.5.

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.

13.1.4 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.

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

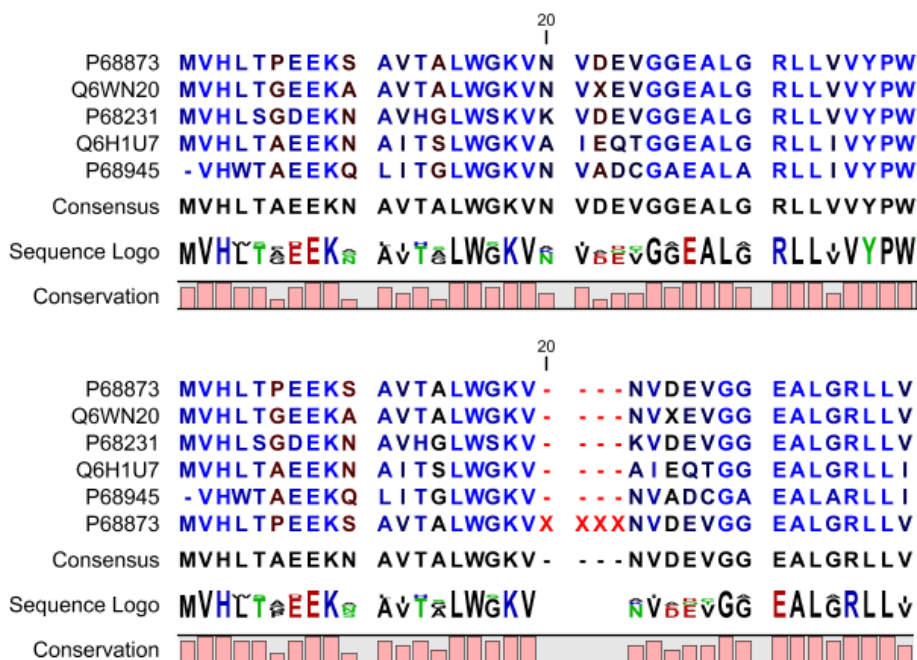


Figure 13.5: The top figures shows the original alignment. In the bottom panel a single sequence with four inserted X's are aligned to the original alignment. This introduces gaps in all sequences of the original alignment. All other positions in the original alignment are fixed.

This will add an annotation labeled "Fixpoint" to the sequence (see figure 13.6). Use this procedure to add fixpoints to the other sequence(s) that should be forced to align to each other.

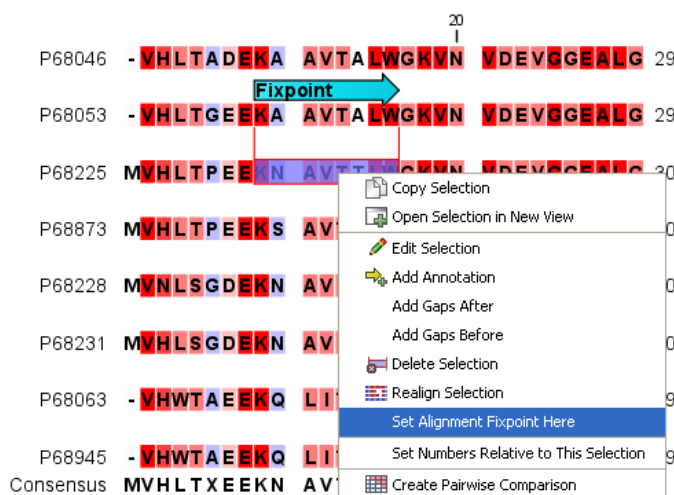


Figure 13.6: Adding a fixpoint to a sequence in an existing alignment. At the top you can see a fixpoint that has already been added.

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 13.7 the result of an alignment using fixpoints is illustrated.

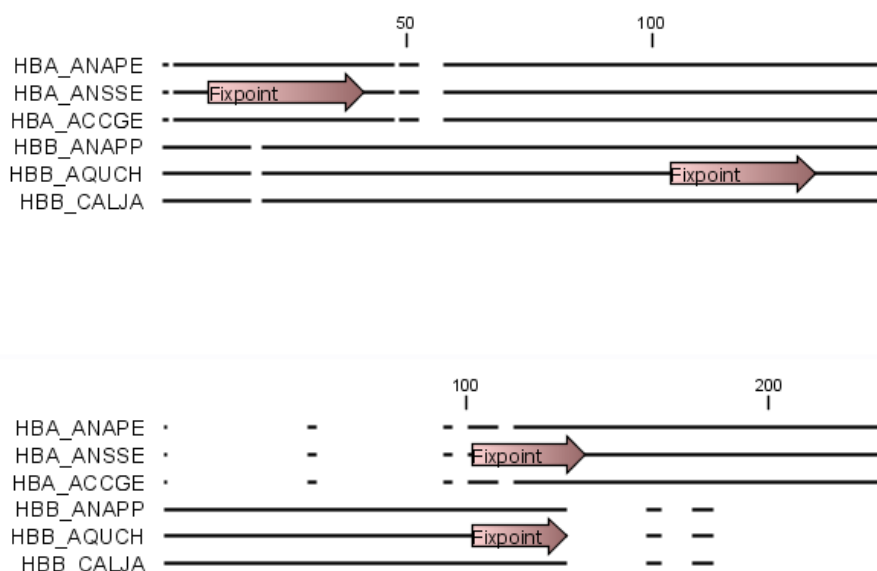


Figure 13.7: *Realigning using fixpoints.* In the top view, fixpoints have been added to two of the sequences. In the view below, the alignment has been realigned using the fixpoints. The three top sequences are very similar, and therefore they follow the one sequence (number two from the top) that has a fixpoint.

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 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

13.2 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 [11.1](#) 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. For example, 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 **G** and **F**. 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 **6.4**.

- **Height** Specifies the height of the graph.
- **Type** The type of the graph: **Line plot**, **Bar plot**, or **Colors**, in which case the graph is seen 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 6.4).
 - **Height** Specifies the height of the graph.
 - **Type** The type of the graph: **Line plot**, **Bar plot**, or **Colors**, in which case the graph is seen 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 13.2.1 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.

13.2.1 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 researchers 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 13.8). 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 13.8). 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 13.8 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.

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 $\log_2 4 = 2 \text{ bits}$ for DNA/RNA and $\log_2 20 \approx 4.32 \text{ bits}$ for proteins.

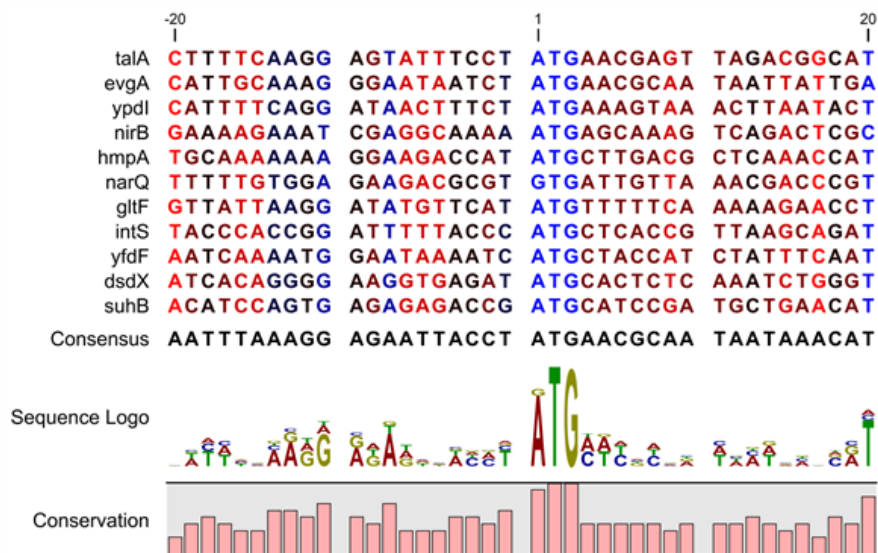


Figure 13.8: Ungapped sequence alignment of eleven *E. coli* sequences defining a start codon. The start codons start at position 1. Below the alignment is shown the corresponding sequence logo. As seen, a GTG start codon and the usual ATG start codons are present in the alignment. This can also be visualized in the logo at position 1.

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 height 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]

13.3 Edit alignments

Move residues and gaps The placement of gaps in the alignment can be changed by modifying the parameters when creating the alignment (see section 13.1). 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 13.9).

Note! Residues can only be moved when they are next to a gap.

```

AGG GAGTCAT      AGG GAGTCAT
AGG GAGTCAT      AGG GAGTCAT
AGG GAGCAGT      AGG GAGCAGT
- - - - -        - - - - -
AGG GTACAGT      AGG GTACAGT
- - - GAGTAGC    - GA G - - TAGC
- - - GAGTAGC    - GA G - - TAGC
- - - GAGTAGG    - GA G - - TAGG
ATG GTGCACC      ATG GTGCACC
ATG GTGCATC      ATG GTGCATC

```

Figure 13.9: Moving a part of an alignment. Notice the change of mouse pointer to a horizontal arrow.

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 five residues for example, 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.

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'

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**.

Copied/transferred annotations will contain the same qualifier text as the original, i.e., the text is not updated. As an example, if the annotation contains 'translation' as qualifier text, this translation will be copied to the new sequence and will thus reflect the translation of the original sequence, and not the new sequence which may differ.

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

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

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 [13.1](#)).

The same procedure can be used for joining two alignments.

13.3.1 Realignment

Realigning a section of an alignment

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 | Choose the option "Realign selection"

This will open a window allowing you to set the parameters for the realignment (see figure 13.10).

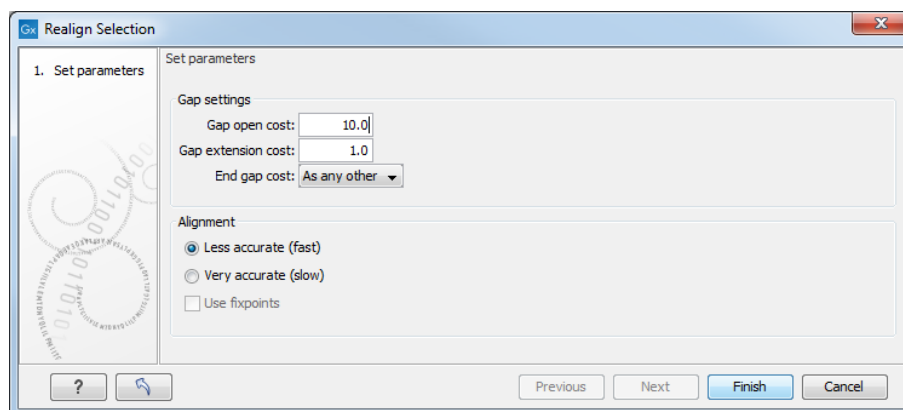


Figure 13.10: *Realigning a section of an alignment.*

Learn more about the options available to you in the sections 13.1.1 and 13.1.2.

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 manually but decide to undo all the edits you just made, you can easily select the region that was edited and realign it.
- **Adjusting the number of gaps.** If you have a region in an alignment which has too many gaps, you can select the region and realign it with a higher gap cost.
- **Combine with fixpoints.** When you have an alignment where two residues are not aligned although they should have been, you can 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.

Realigning a subset of aligned sequences

To realign only a subset of the sequences of an alignment, you have to select the sequences you want to realign by click-and-drag on their entire length, and open the selection in a new view. This means that the sequences you want to select need to be situated on top of each other in a single stack.

A small stack is easily obtainable by dragging the sequences by their names to the top of the alignment. But when a large quantity of sequences needs to be moved to sit together, we recommend using the selection boxes that are available when checking the option "Show selection boxes" from the Alignment settings section in the right-hand side panel (figure 13.11).

Select all the sequences you want to realign independently of the rest of the alignment, and right-click on the name of one of the sequences to choose the option "Sort Sequences by Marked Status". This will bring all checkbox-selected items to the top of the alignment.

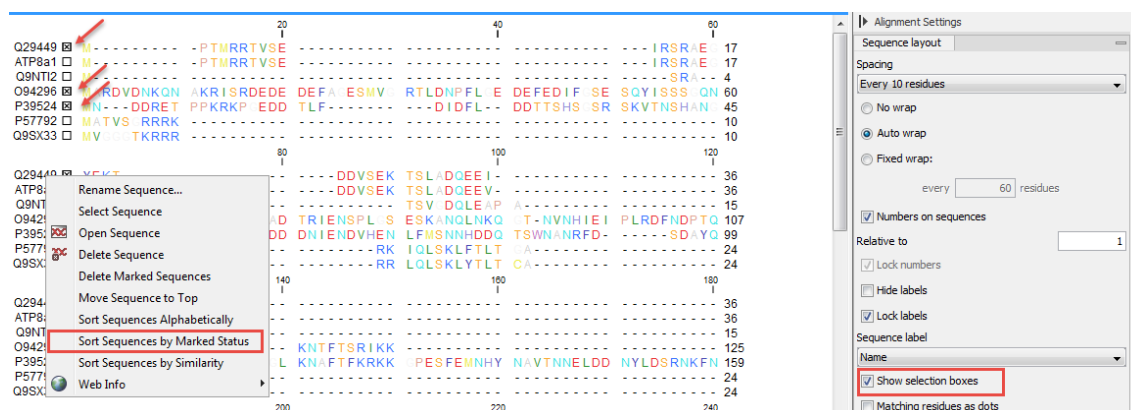


Figure 13.11: Realigning a subset of the sequences that are part of an alignment. Note that we can see here the menu available when right clicking on the names of the sequences.

You can then easily click-and-drag your selection of sequences (this is made easier if you select the "No wrap" setting in the right-hand side panel). By right-clicking on the selected sequences (not on their names, but on the sequences themselves as seen in figure 13.12), you can choose the option "Open selection in a new view", with the ability to run any relevant tool on that sub-alignment.

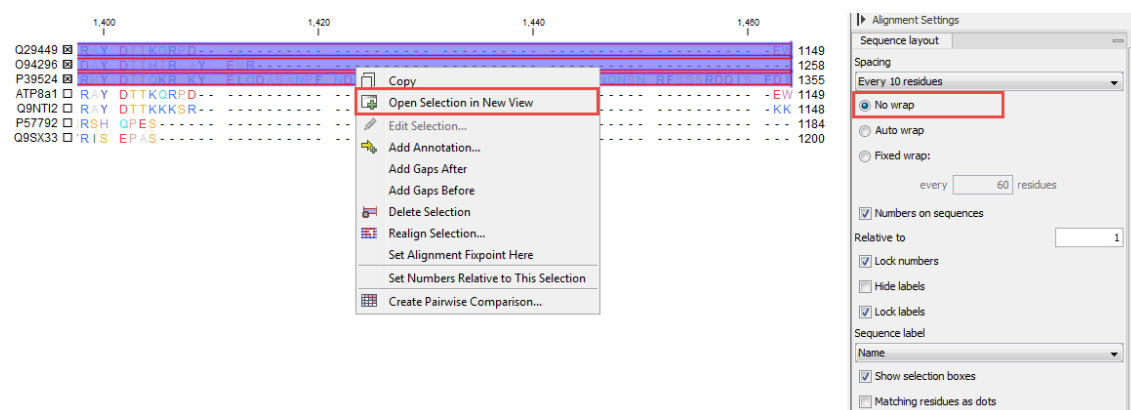


Figure 13.12: Open the selected sequences in a new window to realign them.

13.4 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:

Toolbox | Alignments and Trees (📁) | Join Alignments (🔗)

This opens the dialog shown in figure 13.13.

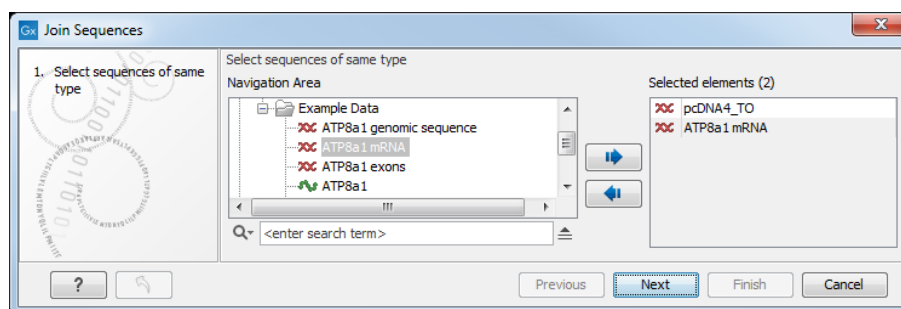


Figure 13.13: Selecting two alignments to be joined.

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 13.14.

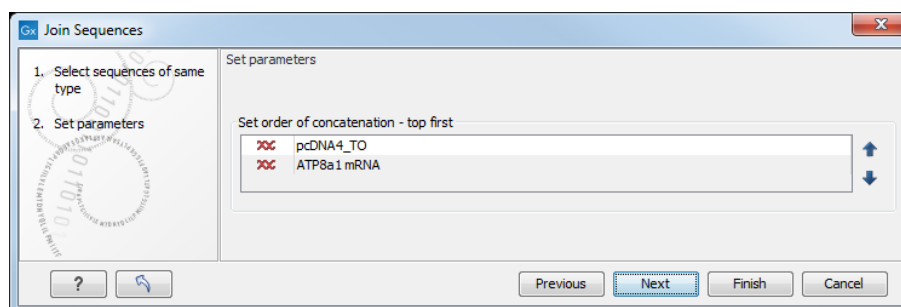


Figure 13.14: Selecting order of concatenation.

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 13.15.

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 13.15 "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.

13.5 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

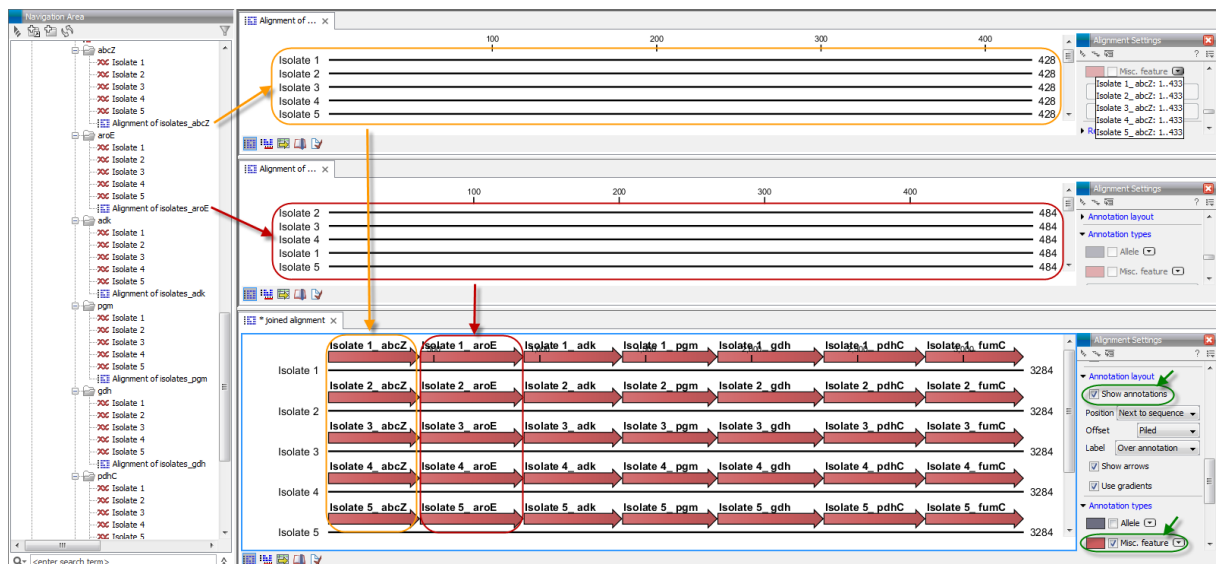


Figure 13.15: The upper part of the figure shows two of the seven alignments for the genes "abcZ" and "aroE" respectively. Each alignment consists of sequences from one gene from five different isolates. The lower part of the figure shows the result of "Join Alignments". Seven genes have been joined 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.

the sequences in the alignment.

In CLC Main Workbench this is done by creating a comparison table:

Toolbox | Alignments and Trees (📄) | Create Pairwise Comparison (📊)

This opens the dialog displayed in figure 13.16:

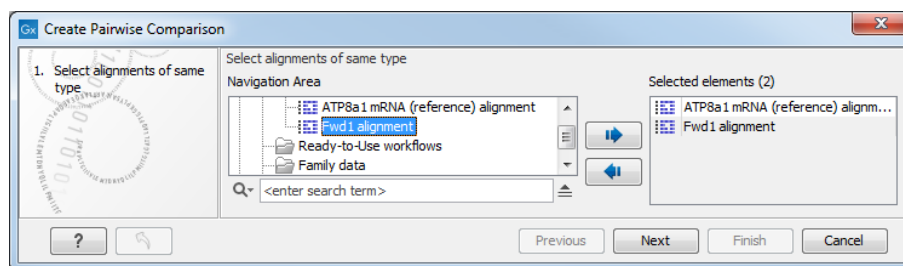


Figure 13.16: Creating a pairwise comparison table.

Select at least two alignments alignment to compare. A pairwise comparison can also be performed for a selected part of an alignment:

right-click on an alignment selection | Pairwise Comparison (📊)

There are five kinds of comparison that can be made between the sequences in the alignment, as shown in figure 13.17.

- **Gaps** Calculates the number of alignment positions where one sequence has a gap and the other does not.

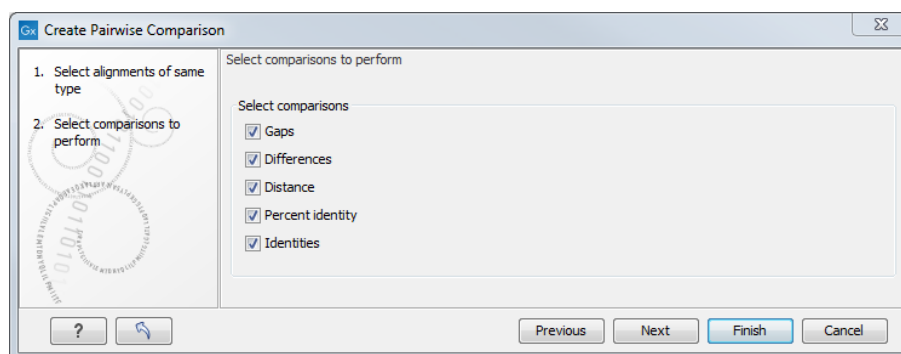


Figure 13.17: Adjusting parameters for pairwise comparison.

- **Identities** Calculates the number of identical alignment positions to overlapping alignment positions between the two sequences. An overlapping alignment position is a position where at least one residue is present, rather than only gaps.
- **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.

13.5.1 The pairwise comparison table

The table shows the results of selected comparisons (see an example in figure 13.18). 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.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Sequence-5	1	1517	1514	1498	1499	1508	1487	1492	1478	1475	1470	1473	1382	1385	1352	1353	1321	1333
Sequence-8	2	7	1517	1499	1500	1509	1488	1493	1479	1476	1471	1474	1384	1388	1353	1354	1320	1332
Sequence-7	3	10	7	1496	1497	1506	1485	1490	1476	1473	1468	1471	1383	1385	1351	1352	1317	1329
Sequence-10	4	26	25	28	1517	1506	1501	1498	1494	1487	1474	1483	1389	1391	1365	1366	1324	1336
Sequence-11	5	25	24	27	7	1507	1500	1499	1491	1488	1477	1484	1390	1392	1363	1364	1323	1335
Sequence-6	6	16	15	18	18	17	1495	1496	1486	1483	1474	1481	1390	1395	1366	1367	1327	1335
Sequence-1	7	37	36	39	23	24	29	1487	1484	1476	1465	1474	1385	1384	1357	1358	1310	1324
Sequence-4	8	32	31	34	26	25	28	37	1479	1494	1498	1474	1379	1385	1355	1356	1319	1326
Sequence-9	9	46	45	48	30	33	38	40	45	1471	1458	1467	1382	1383	1350	1351	1310	1328
Sequence-3	10	49	48	51	37	36	41	48	30	53	1478	1463	1372	1378	1352	1353	1312	1320
Sequence-2	11	54	53	56	50	47	50	59	26	66	46	1452	1367	1372	1350	1351	1315	1326
Sequence-12	12	51	50	53	41	40	43	50	50	57	61	72	1385	1386	1352	1353	1324	1334
Sequence-14	13	142	140	141	135	134	134	139	145	142	152	157	139	1476	1356	1357	1296	1311
Sequence-15	14	139	136	139	133	132	129	140	139	141	146	152	138	48	1357	1358	1301	1315
Sequence-13	15	172	171	173	159	161	158	167	169	174	172	174	172	168	167	1523	1297	1309
Sequence-18	16	171	170	172	158	160	157	166	168	173	171	173	171	167	166	1	1297	1309
Sequence-16	17	203	204	207	200	201	197	214	205	214	212	209	200	228	223	227	227	1466
Sequence-17	18	191	192	195	188	189	188	200	198	196	204	198	190	213	209	215	215	58

Figure 13.18: A pairwise comparison table.

Note that you can change the minimum and maximum values of the gradient coloring by sliding the corresponding cursor along the gradient in the right side panel of the comparison table. The

values that appears when you slide the cursor reflect the percentage of the range of values in the table, and not absolute values.

The following settings are present in the side panel:

- **Contents**

- **Upper comparison** Selects the comparison to show in the upper triangle of the table.
- **Upper comparison gradient** Selects the color gradient to use for the upper triangle.
- **Lower comparison** Selects the comparison to show in the lower triangle. Choose the same comparison as in the upper triangle to show all the results of an asymmetric comparison.
- **Lower comparison gradient** Selects the color gradient to use for the lower triangle.
- **Diagonal from upper** Use this setting to show the diagonal results from the upper comparison.
- **Diagonal from lower** Use this setting to show the diagonal results from the lower comparison.
- **No Diagonal.** Leaves the diagonal table entries blank.

- **Layout**

- **Lock headers** Locks the sequence labels and table headers when scrolling the table.
- **Sequence label** Changes the sequence labels.

- **Text format**

- **Text size** Changes the size of the table and the text within it.
- **Font** Changes the font in the table.
- **Bold** Toggles the use of boldface in the table.

13.5.2 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 13.19) 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.

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 13.19: 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.

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

The method has the inherent drawback that once two sequences are aligned, there is no way of changing their relative alignment based on the information that additional sequences may

contribute later in the process. It is therefore important to make the best possible alignments early in the procedure, to avoid accumulating errors. To accomplish this, a tree of the sequences is usually constructed to guide the progressive alignment algorithm. And to overcome the problem of a time consuming tree construction step, we are using word matching, a method that group sequences in a very efficient way, saving much time, without reducing the resulting alignment accuracy significantly.

Our algorithm (developed by QIAGEN Aarhus) has two speed settings: "standard" and "fast". The **standard method** makes a fairly standard progressive alignment using the fast method of generating a guide tree. When aligning two alignments to each other, two matching columns are scored as the average of all the pairwise scores of the residues in the columns. The gap cost is affine, allowing a different cost for the first gapped position and for the consecutive gaps. This ensures that gaps are not spread out too much.

The **fast method** of alignment uses the same overall method, except that it uses fixpoints in the alignment algorithm based on short subsequences that are identical in the sequences that are being aligned. This allows similar sequences to be aligned much more efficiently, without reducing accuracy very much.

Chapter 14

Phylogenetic trees

Contents

14.1 K-mer Based Tree Construction	263
14.2 Create tree	264
14.3 Model Testing	265
14.4 Maximum Likelihood Phylogeny	267
14.4.1 Bioinformatics explained	270
14.5 Tree Settings	276
14.5.1 Minimap	276
14.5.2 Tree layout	277
14.5.3 Node settings	278
14.5.4 Label settings	279
14.5.5 Background settings	281
14.5.6 Branch layout	281
14.5.7 Bootstrap settings	281
14.5.8 Visualizing metadata	282
14.5.9 Node right click menu	282
14.6 Metadata and phylogenetic trees	285
14.6.1 Table Settings and Filtering	286
14.6.2 Add or modify metadata on a tree	286
14.6.3 Undefined metadata values on a tree	288
14.6.4 Selection of specific nodes	288

Phylogenetics describes the taxonomic 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 [14.4.1](#) 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.

For a given set of aligned sequences (see section 13.1) 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 14.4.1). 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 14.4.1). 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 related, 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 14.4 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 14.3) 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 14.4.1).

14.1 K-mer Based Tree Construction

The K-mer Based Tree Construction tool 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 14.1):

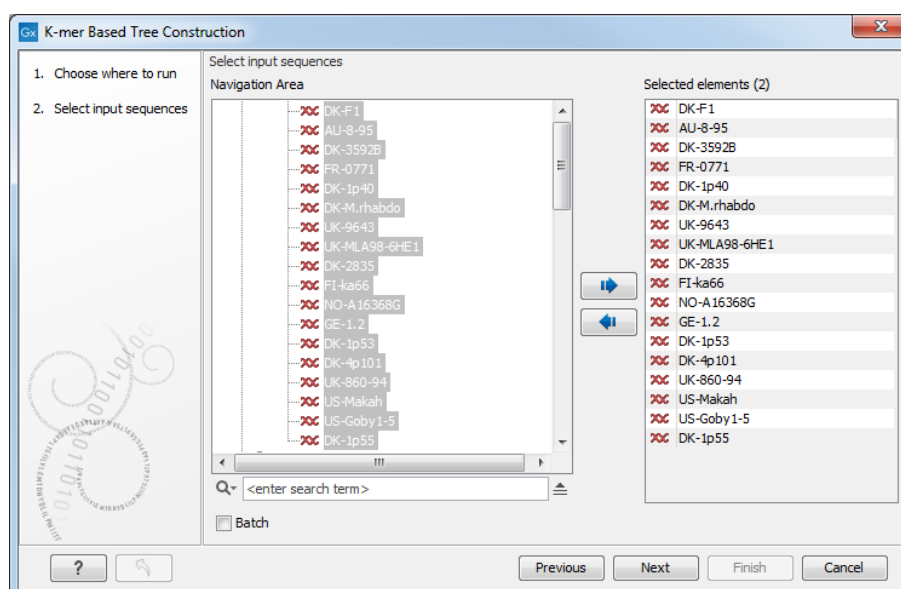


Figure 14.1: Select sequences needed for creating a tree with K-mer based tree construction.

Next, select the construction method, specify the k-mer length and select a distance measure for tree construction (figure 14.2):

- **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 14.4.1):
 - * The **UPGMA** method. Assumes constant rate of evolution.
 - * The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.
- **K-mer settings**

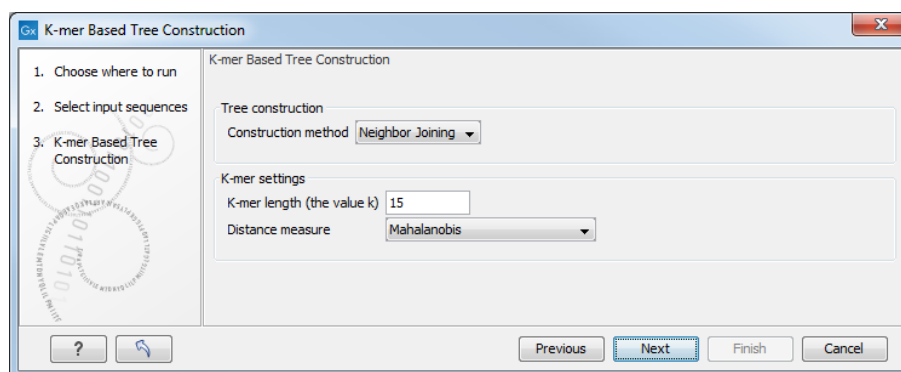


Figure 14.2: Select the construction method, and specify the k-mer length and a distance measure.

- **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 14.4.1 for further details.

14.2 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 14.3:

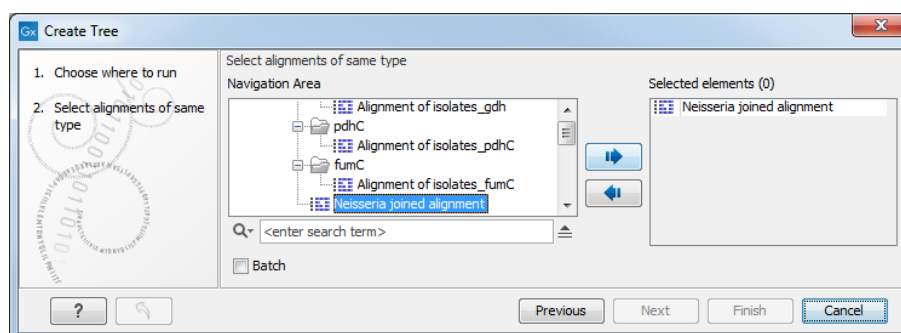


Figure 14.3: Creating a tree.

If an alignment was selected before choosing the Toolbox action, this alignment is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove elements from the **Navigation Area**. Click **Next** to adjust parameters.

Figure 14.4 shows the parameters that can be set for this distance-based tree creation:

- Tree construction (see section 14.4.1)
 - Tree construction method
 - * The **UPGMA** method. Assumes constant rate of evolution.
 - * The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.

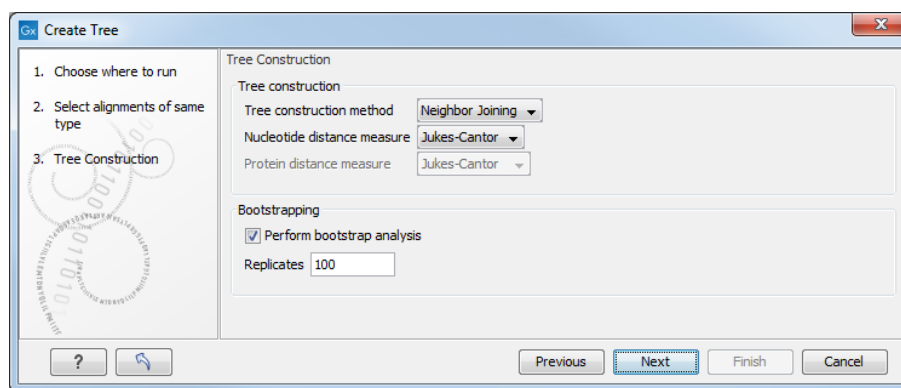


Figure 14.4: Adjusting parameters for distance-based methods.

- 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 14.4.1). 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 14.4.1.

14.3 Model Testing

As the Model Testing tool can help identify the best substitution model (14.4.1) to be used for Maximum Likelihood Phylogeny tree construction, it is recommended to run 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 14.5):

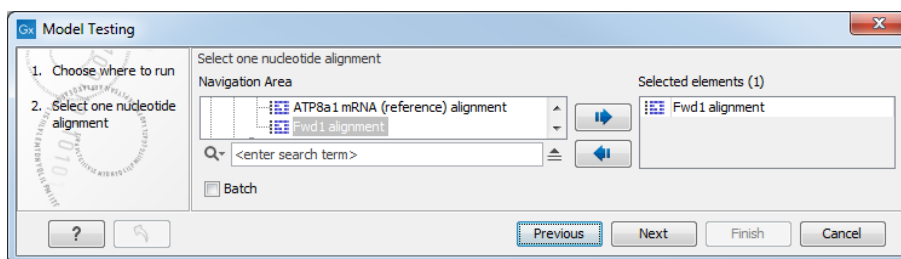


Figure 14.5: Select alignment for model testing.

Specify the parameters to be used for model testing (figure 14.6):

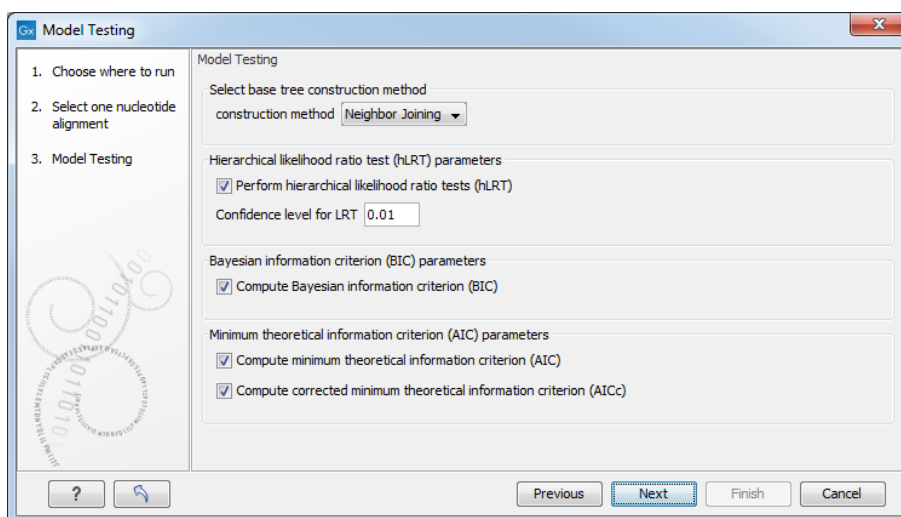


Figure 14.6: Specify parameters for model testing.

- **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:
 - * 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 $BIC = -2\ln(L) + K\ln(n)$, 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 $AIC = -2\ln(L) + 2K$, 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 (AICc)** Rank substitution models based on minimum corrected theoretical information criterion (AICc). Formula used is $AICc = -2\ln(L) + 2K + 2K(K+1)/(n-K-1)$, 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.

14.4 Maximum Likelihood Phylogeny

To generate a maximum likelihood based phylogenetic tree:

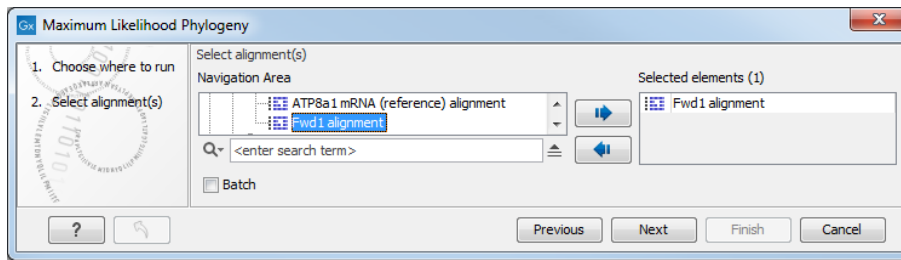


Figure 14.7: Select the alignment for tree construction.

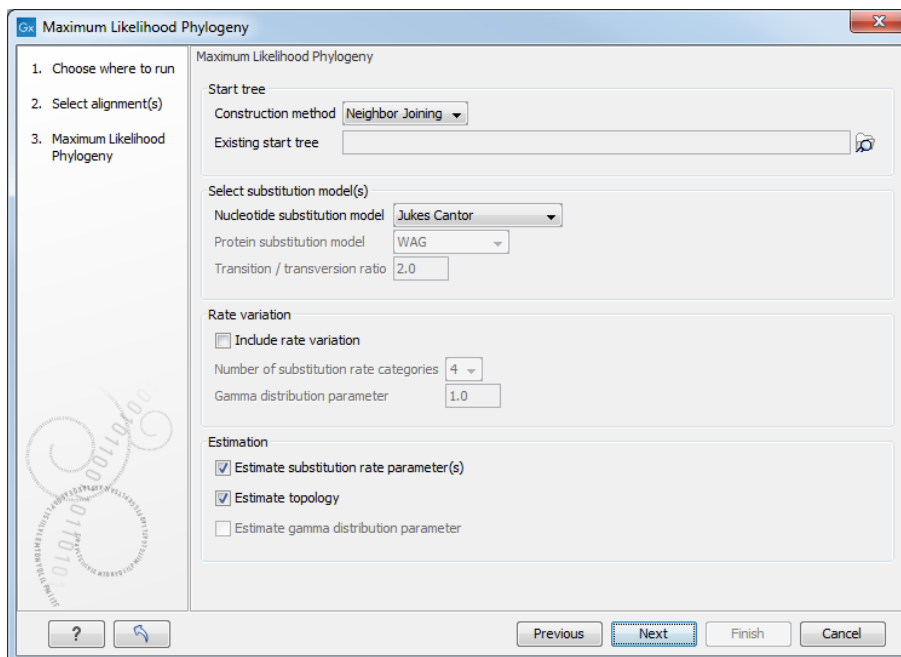


Figure 14.8: Adjusting parameters for maximum likelihood phylogeny

Toolbox | Alignments and Trees (📁) | Maximum Likelihood Phylogeny (⚙️)

The following parameters can be set for the maximum likelihood based phylogenetic tree (see figure 14.8):

- **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
- **Existing 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**

- **Nucleotide 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 transition/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 14.4.1.

- **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 14.9).

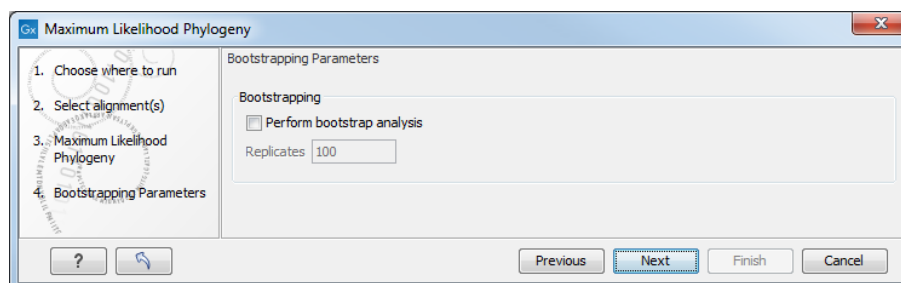


Figure 14.9: Adjusting parameters for ML phylogeny

- **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 14.4.1). 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.

14.4.1 Bioinformatics explained

The phylogenetic tree

The evolutionary hypothesis of a phylogeny can be graphically represented by a phylogenetic tree.

Figure 14.10 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 taxonomic units. In this case, the terminal nodes (also called leaves or tips of the tree) represent extant species of *Hominidae* and are the *operational taxonomic units* (OTUs). The internal nodes, which here represent extinct common ancestors of the great apes, are termed *hypothetical taxonomic units* since they are not directly observable.

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 14.10 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

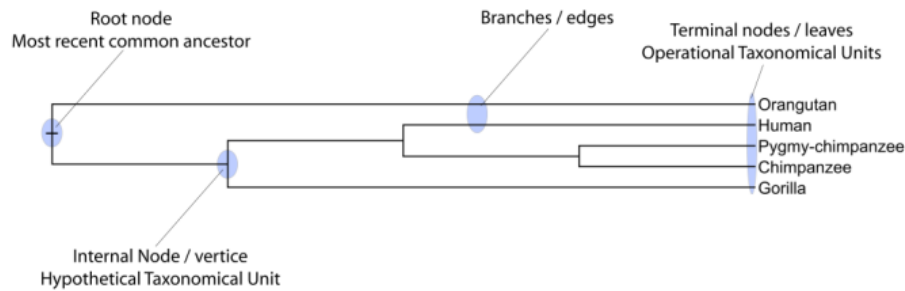


Figure 14.10: A proposed phylogeny of the great apes (*Hominidae*). Different components of the tree are marked, see text for description.

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 were 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) occurring 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 14.4.1).

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.

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 reconstructs 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 neighbor 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 lengths. 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 $P(\text{tree}|\text{data})$ 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 $P(\text{tree}|\text{data})$ so instead ML based methods have to compute the probability of the data given a tree, i.e. $P(\text{data}|\text{tree})$. 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. To help users choose the correct model for a data set, the Model Testing tool (see section 14.3) can be used to test a range of different models for input nucleotide sequences.

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 14.3) 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 neighbor-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 resampling 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.

14.5 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 following section describes the visualization options available from the Tree Settings side panel. Note however that editing legend boxes related to metadata can be done directly from editing the metadata table (see section 14.6).

The preferred tree layout settings (user defined tree settings) can be saved and applied via the top right **Save Tree Settings** (figure 14.11). 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.

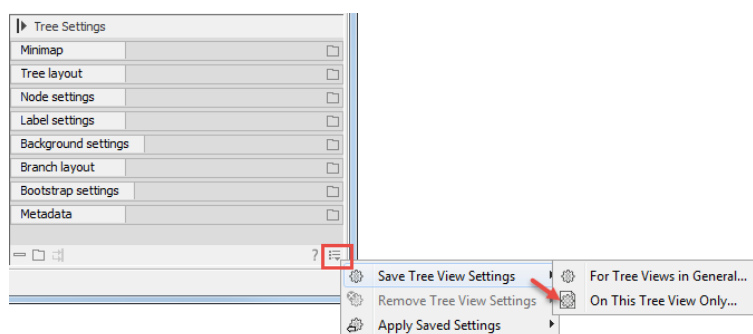


Figure 14.11: Save, remove or apply preferred layout settings.

14.5.1 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 14.12). To navigate the tree using the Minimap, click on the Minimap with the mouse and move the grey square around within the Minimap.

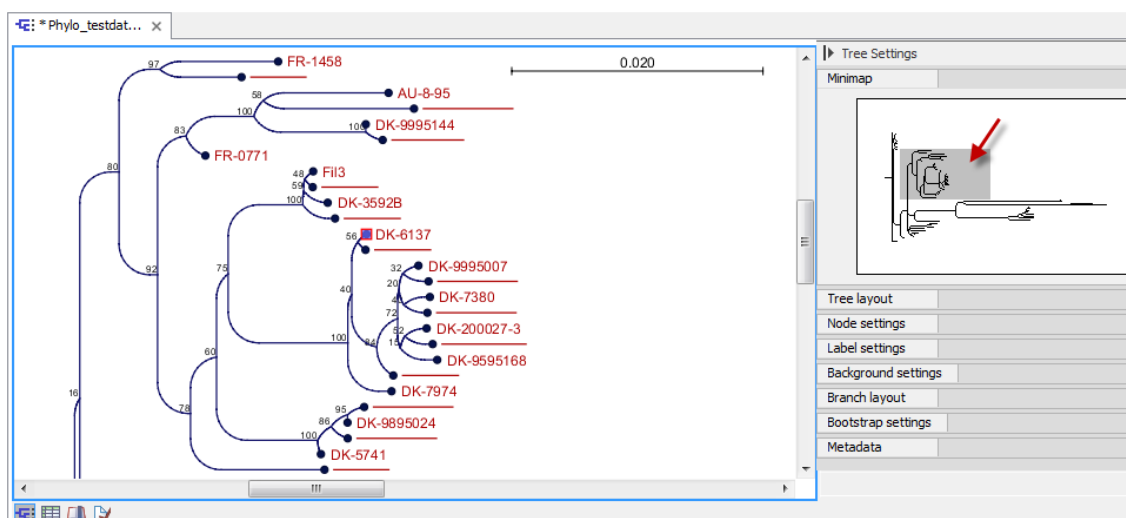


Figure 14.12: Visualization of a phylogenetic tree. The grey square in the Minimap shows the part of the tree that is shown in the View Area.

14.5.2 Tree layout

The **Tree Layout** can be adjusted in the Side Panel (figure 14.13).

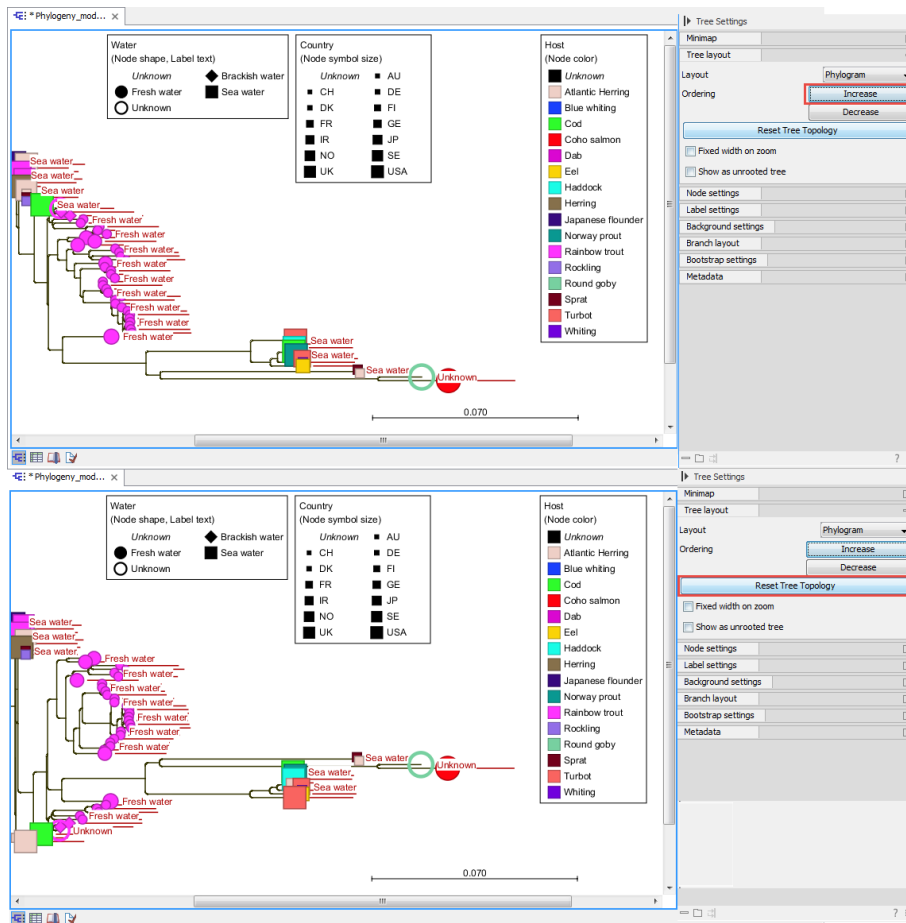


Figure 14.13: The tree layout can be adjusted in the Side Panel. The top part of the figure shows a tree with increasing node order. In the bottom part of the figure the tree has been reverted to the original tree topology.

- **Layout** Selects one of the five layout types: Phylogram, Cladogram, Circular Phylogram, Circular Cladogram or Radial. Note that only the Cladogram layouts are available if all branches in the tree have zero length.
 - **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 14.13) or **Decreasing**.

- **Reset Tree Topology** Resets to the default tree topology and node order (see figure 14.13).
- **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.

14.5.3 Node settings

The nodes can be manipulated in several ways.

- **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 14.14.

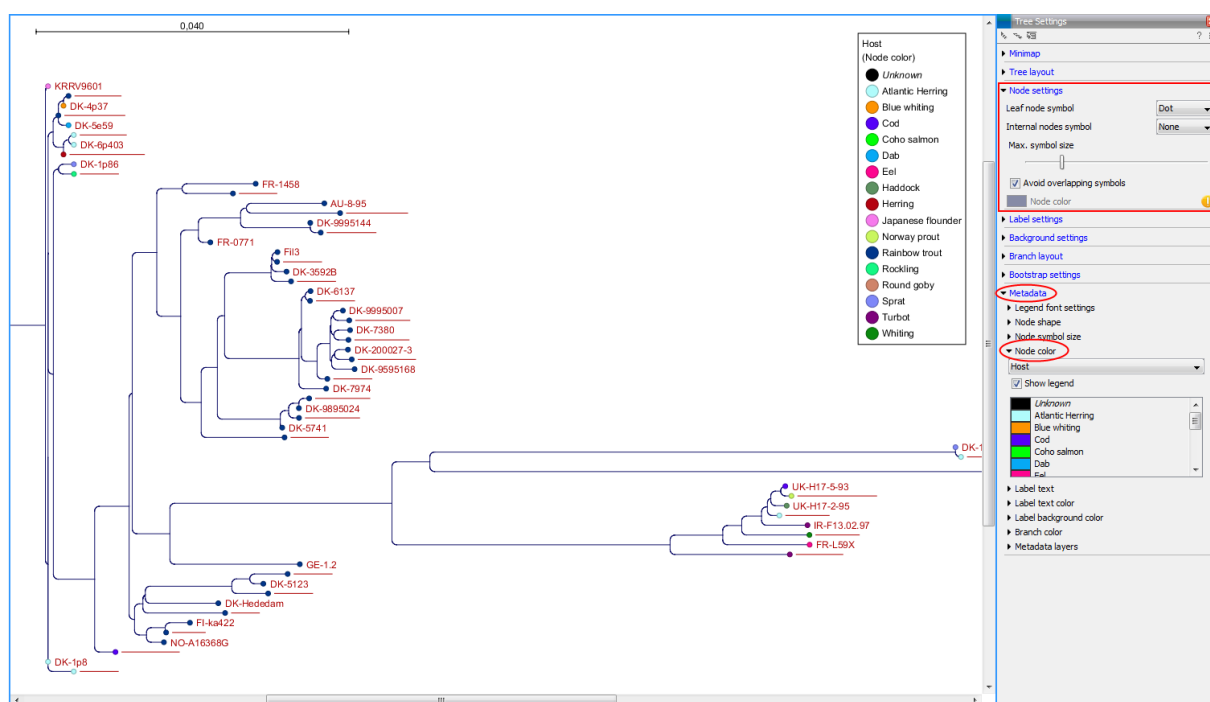


Figure 14.14: The Node Layout settings. Node color is specified by metadata and is therefore inactive in this example.

14.5.4 Label settings

- **Label font settings** Can be used to specify/adjust font type, size and typography (Bold, Italic or normal).
- **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 14.15).
- **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 14.5.9).
- **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.

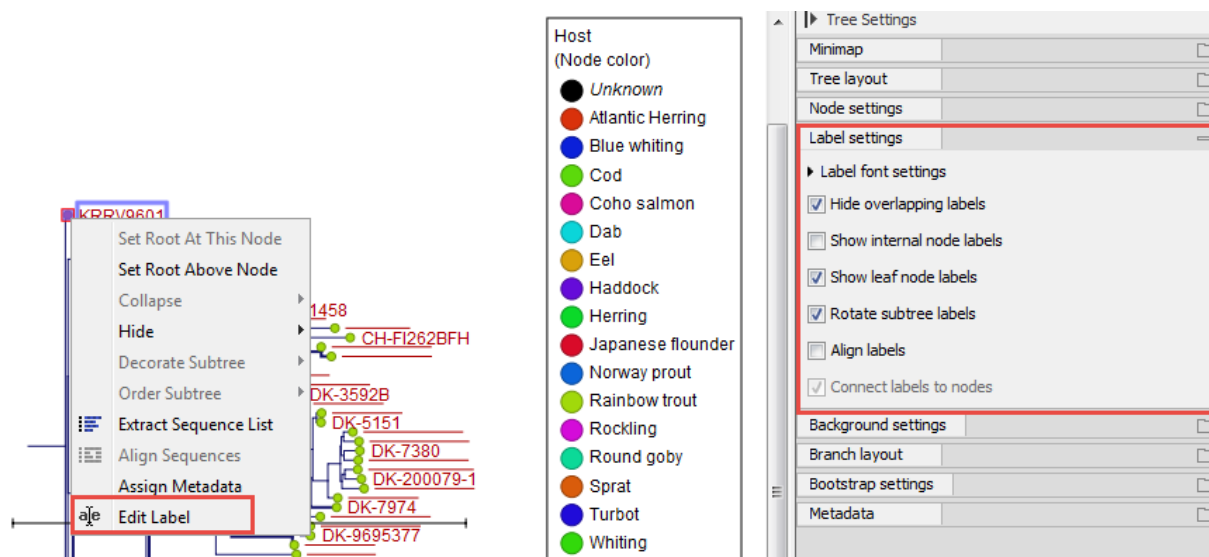


Figure 14.15: "Edit label" in the right click menu can be used to customize the label text. The way node labels are displayed can be controlled through the labels settings in the right side panel.

When working with big trees there is typically not enough space to show all labels. As illustrated in figure 14.15, only some of the labels are shown. The hidden labels are illustrated with thin horizontal lines (figure 14.16).

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 14.16 and figure 14.17). The last option is to disable **Hide overlapping labels** under "Label settings" in the right side panel. When this option is unchecked

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 14.5.5).

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.

Note! The text within labels can be edited by editing the metadata table values directly.

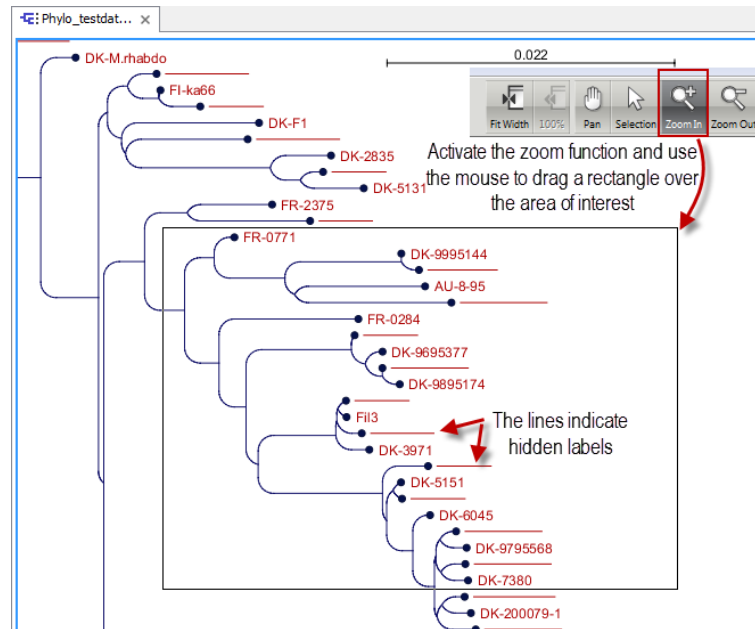


Figure 14.16: The zoom function in the upper right corner of CLC Genomics Workbench can be used to zoom in on a particular region of the tree. When the zoom function has been activated, use the mouse to drag a rectangle over the area that you wish to zoom in at.

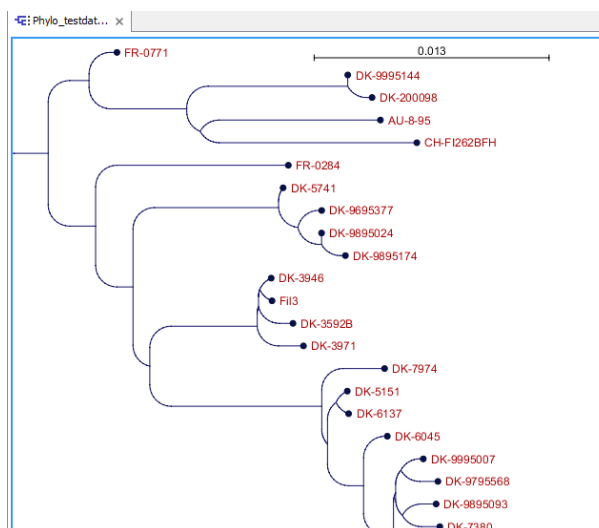


Figure 14.17: After zooming in on a region of interest more labels become visible. In this example all labels are now visible.

14.5.5 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.

14.5.6 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.
- **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 14.18.

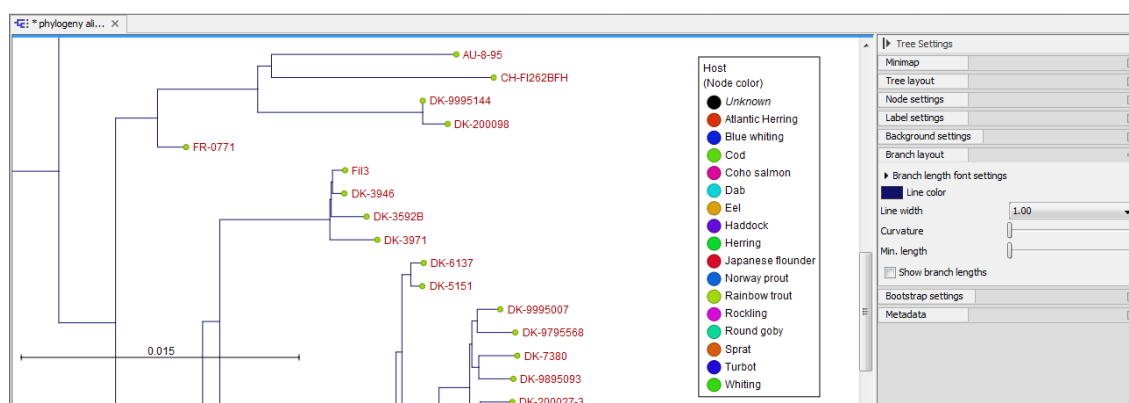


Figure 14.18: Branch Layout settings.

14.5.7 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 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.

Some branches in rooted trees may not have bootstrap values. Trees constructed with neighbour joining are unrooted and to correctly visualize them, the "Radial" view is required. In all other tree views we need a root to visualize the tree. An "artificial node" and therefore an extra branch are created for such visualization to achieve this, which makes it look like a bootstrap value is missing

- **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.

14.5.8 Visualizing metadata

Metadata associated with a phylogenetic tree (described in detail in section 14.6) 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 14.19.
- **Label text color** The label text can be colored and used to visualize metadata (see figure 14.19).
- **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 14.14).

14.5.9 Node right click menu

Additional options for layout and extraction of subtree data are available when right clicking the nodes (figure 14.15):

- **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.

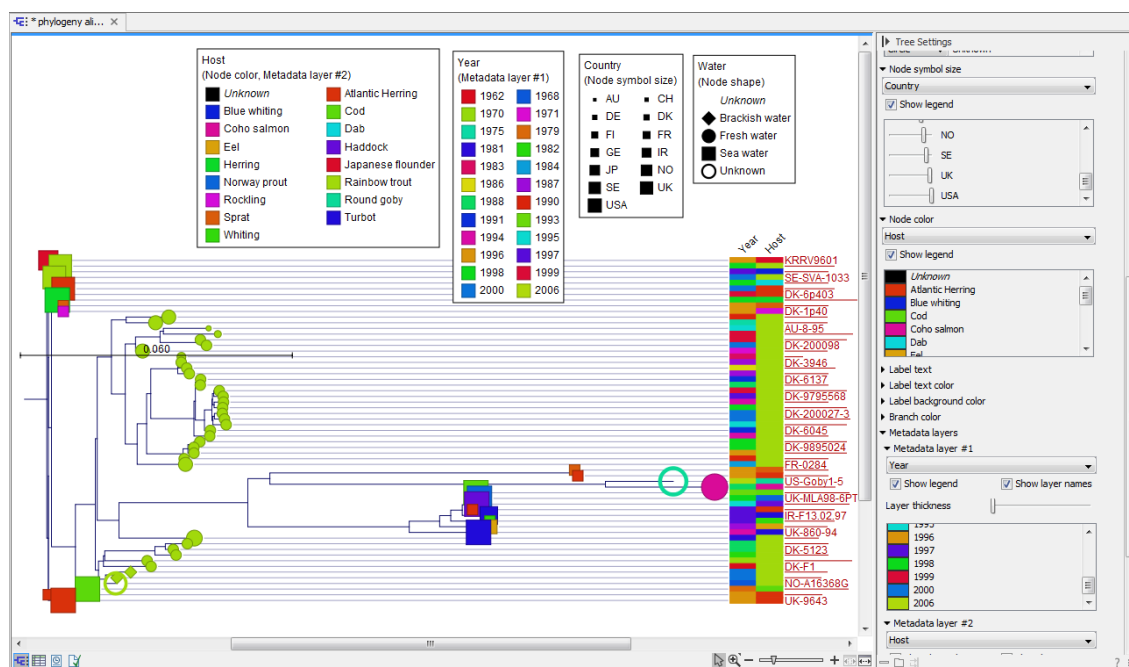


Figure 14.19: Different types of metadata can be visualized by adjusting node size, shape, and color. Two color-code metadata layers (Year and Host) are shown in the right side of the tree.

- 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.
- 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 14.20). When hiding nodes, a new button appears labeled "Show X hidden nodes" in the Side Panel under "Tree Layout" (figure 14.21). 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. To save the decoration, see figure 14.11 and use option: **Save/Restore Settings | Save Tree View Settings On This Tree View only**.
- 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

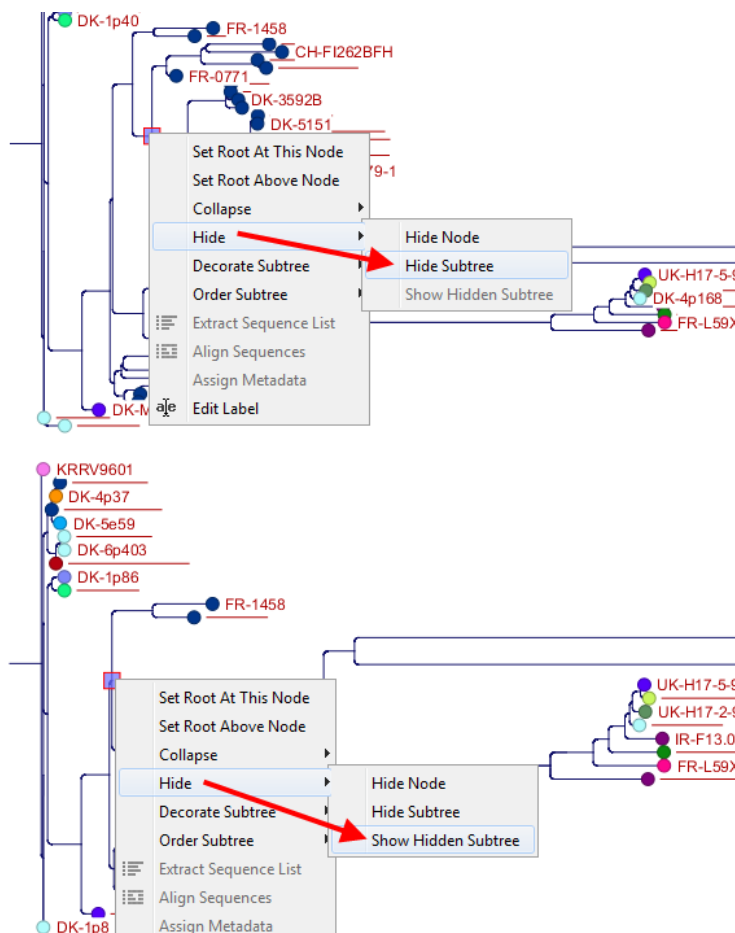


Figure 14.20: A subtree can be hidden by selecting "Hide Subtree" and is shown again when selecting "Show Hidden Subtree" on a parent node.

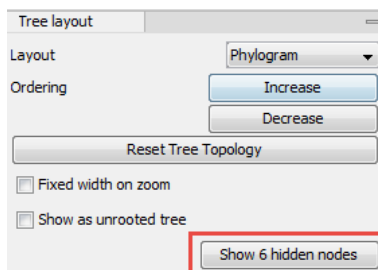


Figure 14.21: When hiding nodes, a new button labeled "Show X hidden nodes" appears in the Side Panel under "Tree Layout". When pressing this button, all hidden nodes are brought back.



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**

14.6 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 14.22).

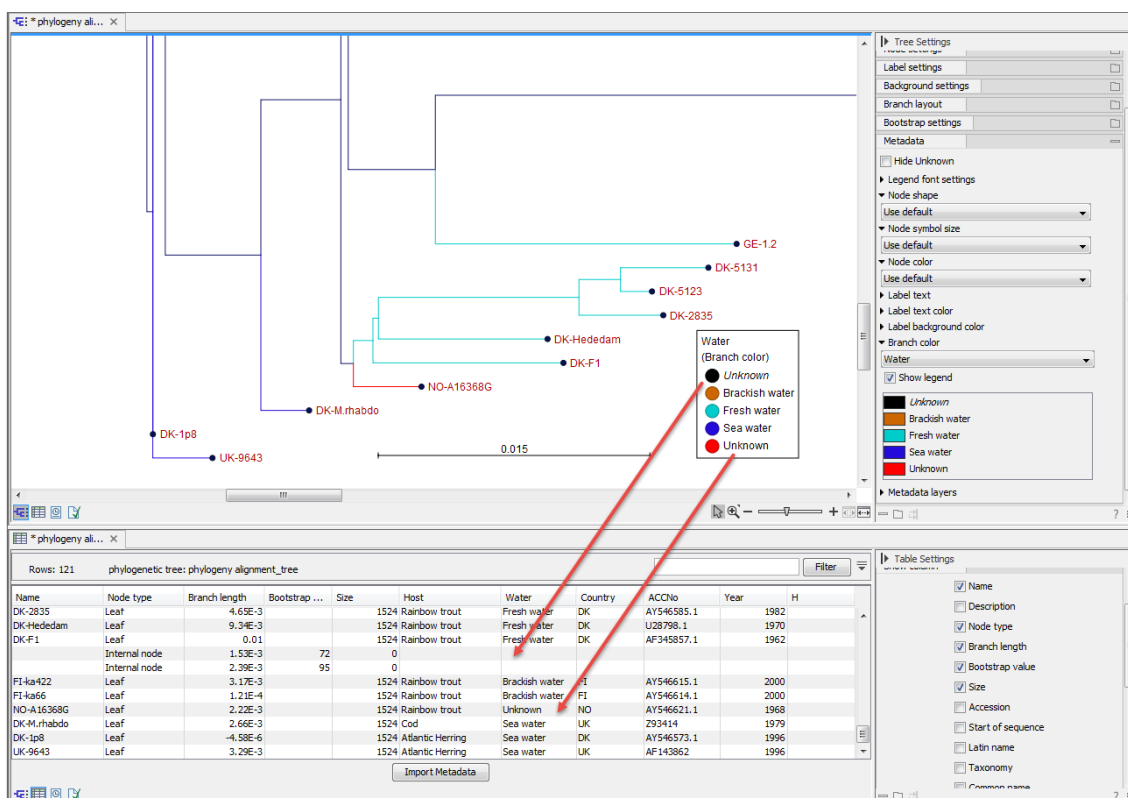


Figure 14.22: Tabular metadata that is associated with an existing tree shown in a split view. Note that *Unknown* written in italics (black branches) refer to missing metadata, while *Unknown* in regular font refers to metadata labeled as "Unknown".

Additional metadata can be associated with a tree by clicking the **Import Metadata** button. This will open up the dialog shown in figure 14.23.

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 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 14.23. 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.

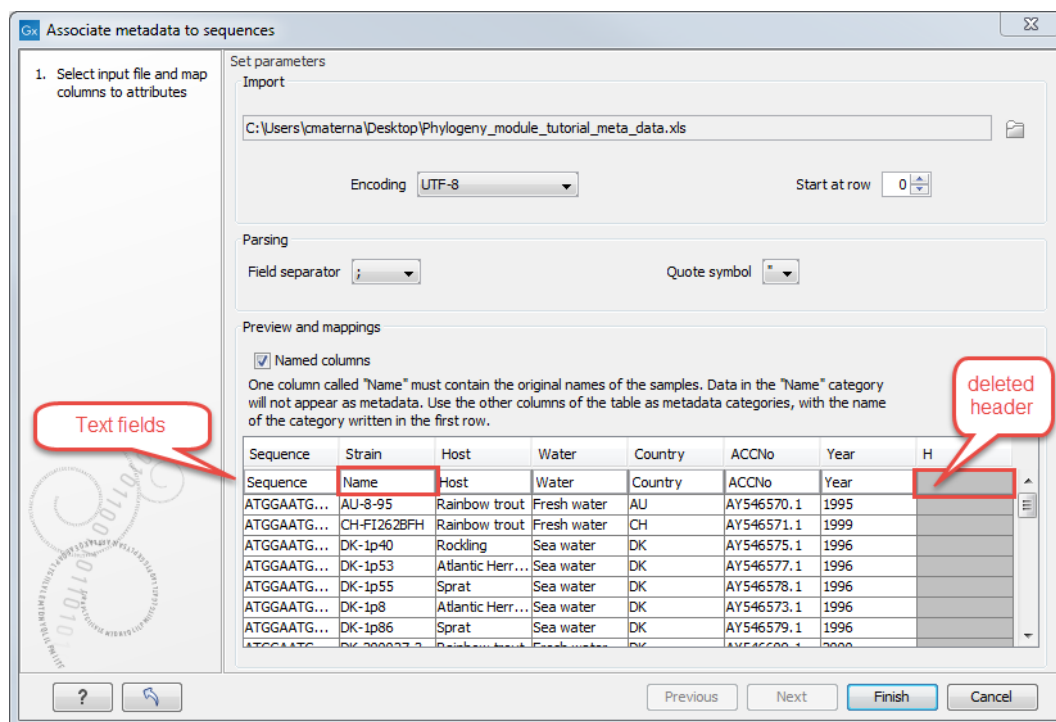


Figure 14.23: Import of metadata for a tree. The second column named "Strain" is chosen as the common denominator by entering "Name" in the text field of the column. The column labeled "H" is ignored by not assigning a column heading to this column.

14.6.1 Table Settings and Filtering

How to use the metadata table (see figure 14.24):

- **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).

14.6.2 Add or modify metadata on a tree

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 14.25):

Figure 14.24: 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 14.25: Right click options in the metadata table.

- Assign Metadata** The right click option "Assign Metadata" can be used for four purposes.
 - 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". 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" (see an example in figure 14.26). This will change values in all selected rows in the column that was clicked.

Size	Host	Water	Country	ACCNo
1524	Round goby	Unknown	USA	AB672615.1
1524	Rainbow trout	Fresh water	CH	AY546571.1
1524	Eel	Sea water	FR	AY546618.1
0	0			
1524	Rainbow trout			U28800
1524	Rainbow trout			AF345857.1
1524	Rainbow trout			AY546570.1
0	0			
1524	Rainbow trout			AF143863
1524	Rainbow trout			U28798.1
0	0			
1524	Whiting	Sea water	DK	AY546581.1

Figure 14.26: To modify existing metadata, click on the specific field, select "Edit <column header>" and provide a new value.

14.6.3 Undefined metadata values on a tree

When visualizing a metadata category where one or more nodes in the tree have undefined values (empty fields in the table), these nodes will be visualized using a default value in **italics** in the top of the legend (see the entry "Unknown" in figure 14.27). To remove this entry in the legend, all nodes must have a value assigned in the corresponding metadata category.

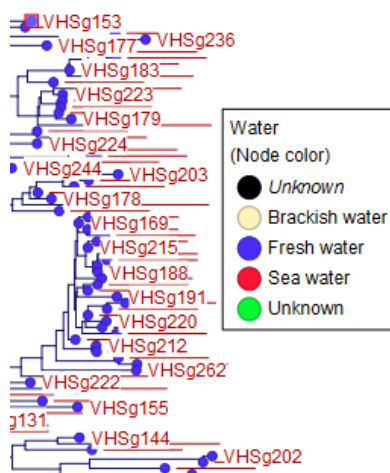


Figure 14.27: A legend for a metadata category where one or more values are undefined. Fill your metadata table with a value of your choice to edit the mention of ("Unknown" in the legend. Note that the "Unknown" that is not in italics is used for data that had a value written as "Unknown" in the metadata table.

14.6.4 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 14.28).
- *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.
- *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 13.5.2). Next, change relevant option in the multiple alignment wizard that pops up and click **Finish**. The multiple alignment will now be generated.

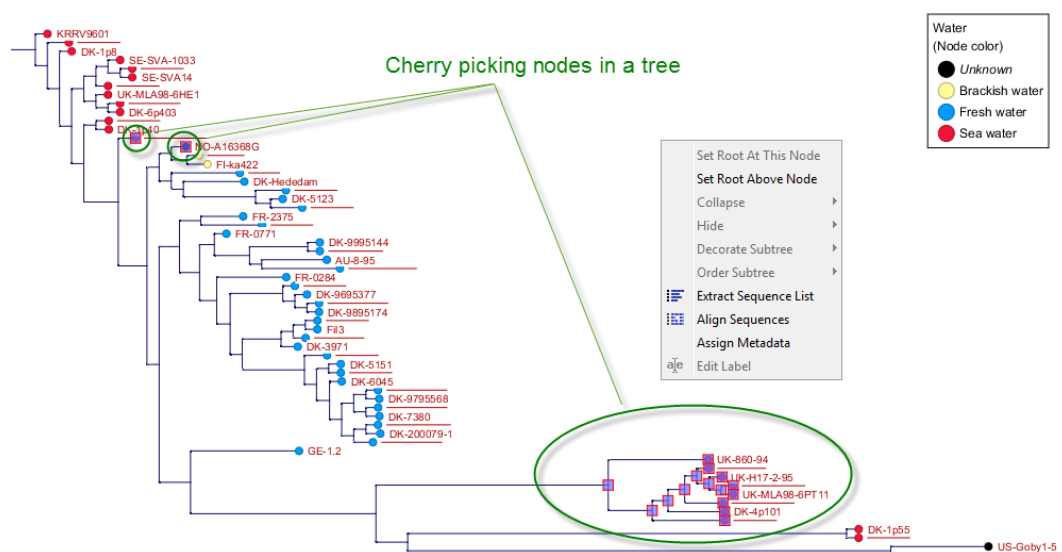


Figure 14.28: *Cherry picking nodes in a tree. The selected leaf sequences can be extracted by right clicking on one of the selected nodes and selecting "Extract Sequence List". It is also possible to Align Sequences directly by right clicking on the nodes or leaves.*

Chapter 15

General sequence analyses

Contents

15.1 Extract Annotations	290
15.2 Extract sequences	292
15.3 Shuffle sequence	294
15.4 Dot plots	295
15.4.1 Create dot plots	296
15.4.2 View dot plots	296
15.4.3 Bioinformatics explained: Dot plots	297
15.4.4 Bioinformatics explained: Scoring matrices	300
15.5 Local complexity plot	304
15.6 Sequence statistics	304
15.6.1 Bioinformatics explained: Protein statistics	307
15.7 Join sequences	309
15.8 Pattern discovery	310
15.8.1 Pattern discovery search parameters	311
15.8.2 Pattern search output	312
15.9 Motif Search	312
15.9.1 Dynamic motifs	312
15.9.2 Motif search from the Toolbox	314
15.9.3 Java regular expressions	316
15.10 Create motif list	317

CLC Main Workbench offers different kinds of sequence analyses that apply to both protein and DNA.

The analyses are described in this chapter.

15.1 Extract Annotations

The **Extract annotations** tool makes it very easy to extract parts of a sequence (or several sequences) based on its annotations. In just a few steps, it becomes possible to:

- extract all tRNA genes from a genome.
- automatically add flanking regions to the annotated sequences.
- search for specific words in all available annotations.
- extract nucleotide sequences of differentially expressed genes or transcripts when using RNA-seq statistical comparisons as input.

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, go to:

Toolbox | Utility Tools | Extract Annotations (➡)

This opens the dialog shown in figure 24.9 that allows specification of which sequence to extract annotations from.

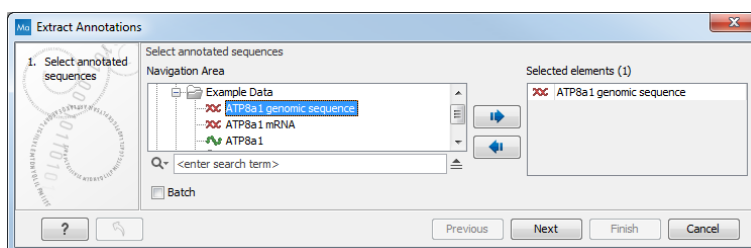


Figure 15.1: Select one or more sequences to extract annotations from.

Click **Next**. At the top of the dialog shown in figure 24.10 you can specify which annotations to use:

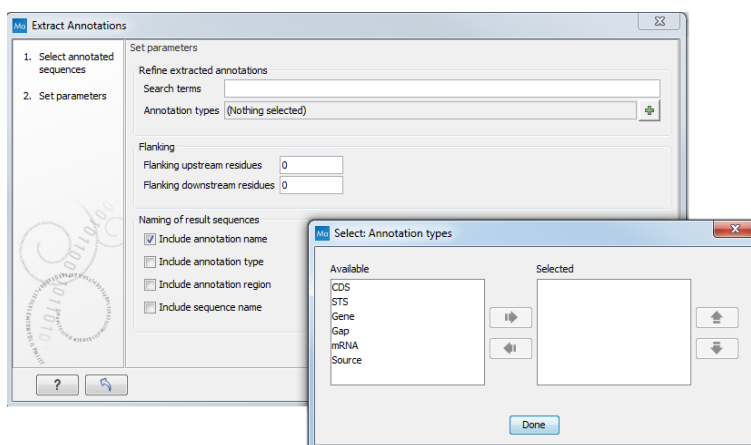


Figure 15.2: Adjusting parameters for extract annotations.

- **Search terms.** 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. For example, if you have a gene 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: "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 **Finish** to start the tool.

15.2 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 on an entire file of one of the above types, **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 18.7.6 that describes the function "Extract from Selection" that also can be selected from the right click menu (see figure 15.3).

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, by going to:

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 15.3.

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.

"Note also, that paired reads will be extracted in accordance with the read group settings, which is specified during the original import of the reads. If the orientation has since been changed (e.g. using the Element Info tab for the sequence list) the read group information will be modified and reads will be extracted as specified by the modified read group. The default read group orientation is forward-reverse."

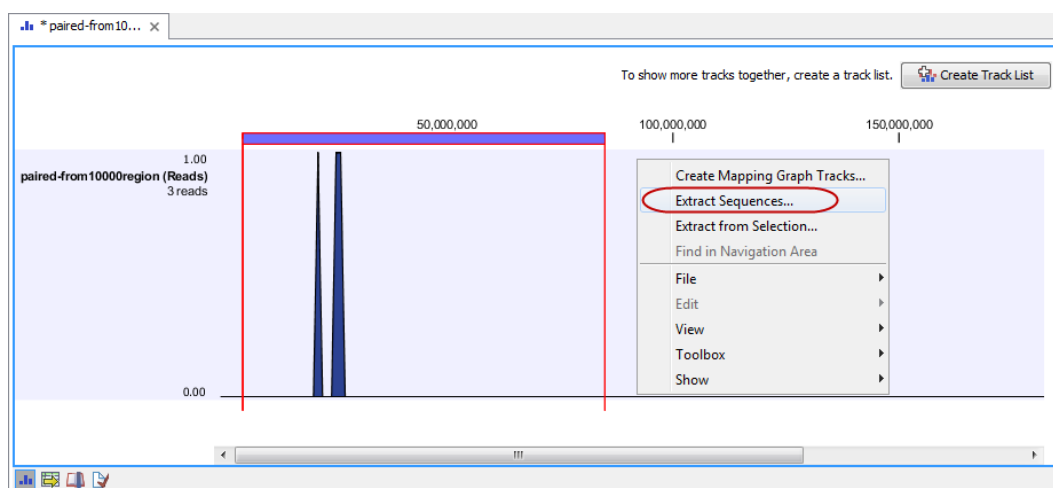


Figure 15.3: Right click somewhere in the reads track area and select "Extract Sequences".

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 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

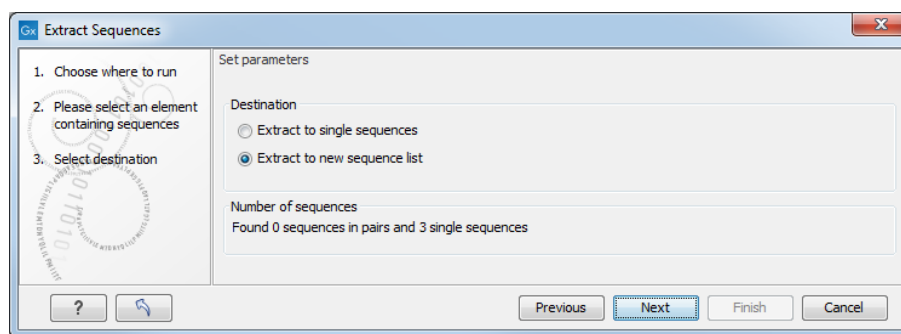


Figure 15.4: Choosing whether the extracted sequences should be placed in a new list or as single sequences.

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.

15.3 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. To launch the tool, go to:

Toolbox | **General Sequence Analysis** (📁) | **Shuffle Sequence** (🔀)

Choose a sequence for shuffling. If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists, from the selected elements.

Click **Next** to determine how the shuffling should be performed.

In this step, shown in figure 15.5:

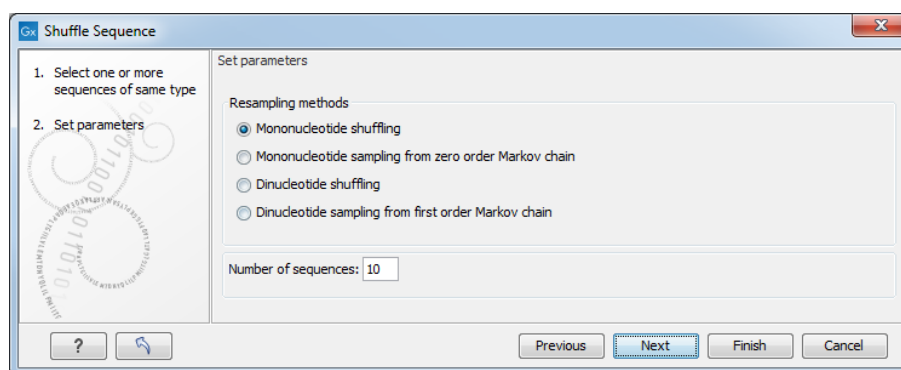


Figure 15.5: Parameters for shuffling.

For nucleotides, the following parameters can be set:

- **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 **Finish** to start the tool.

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.

15.4 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.

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, various substitution matrices can be applied in order to take the evolutionary distance of the two sequences into account.

15.4.1 Create dot plots

To create a dot plot, go to:

Toolbox | General Sequence Analysis (📁) | Create Dot Plot (🔗)

In the dialog that opens, select a sequence and click **Next** to adjust dot plot parameters (figure 15.6).

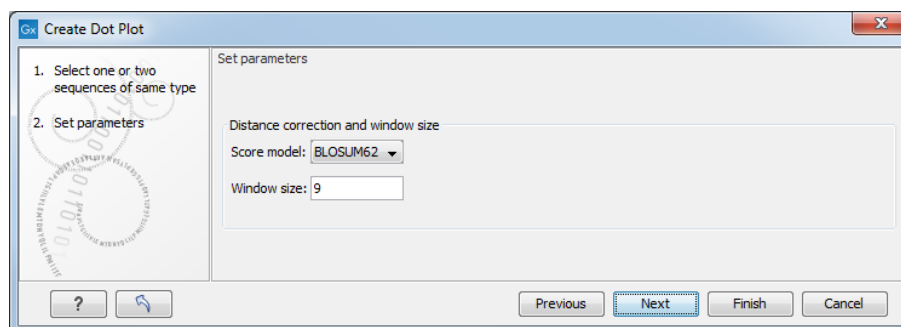


Figure 15.6: Setting the dot plot parameters.

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'.

Note! Calculating dot plots takes up a considerable amount of memory in the computer. Therefore, you will see a warning message if the sum of the number of nucleotides/amino acids in the sequences is higher than 8000. If you insist on calculating a dot plot with more residues the Workbench may shut down, but still allowing you to save your work first. However, this depends on your computer's memory configuration.

Click **Finish** to start the tool.

15.4.2 View dot plots

A view of a dot plot can be seen in figure 15.7. 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.

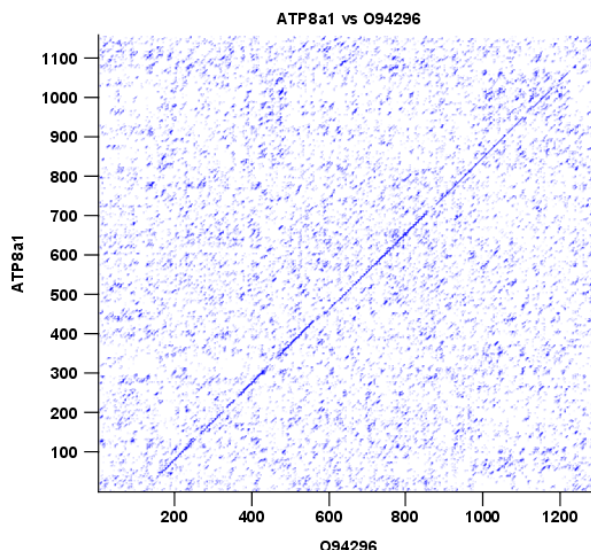


Figure 15.7: A view is opened showing the dot plot.

Adjusting the sliders above the gradient box is also practical, when producing an output for printing. (Too much background color might not be desirable). By crossing one slider over the other (the two sliders change side) the colors are inverted, allowing for a white background. (If you choose a color gradient, which includes white). See figure 15.7.

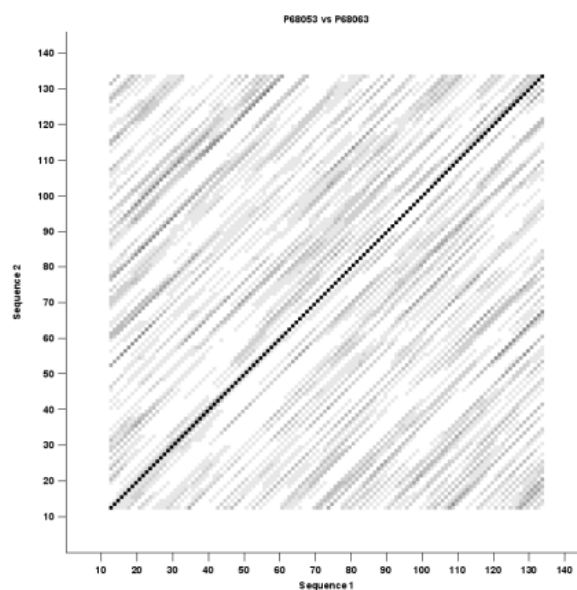


Figure 15.8: Dot plot with inverted colors, practical for printing.

15.4.3 Bioinformatics explained: Dot plots

Dot plots are two-dimensional plots where the x-axis and y-axis each represents a sequence and the plot itself shows a comparison of these two sequences by a calculated score for each position of the sequence. If a window of fixed size on one sequence (one axis) match to the other sequence a dot is drawn at the plot. Dot plots are one of the oldest methods for comparing two sequences [Maizel and Lenk, 1981].

The scores that are drawn on the plot are affected by several issues.

- Scoring matrix for distance correction.
Scoring matrices (BLOSUM and PAM) contain substitution scores for every combination of two amino acids. Thus, these matrices can only be used for dot plots of protein sequences.
- Window size
The single residue comparison (bit by bit comparison(window size = 1)) in dot plots will undoubtedly result in a noisy background of the plot. You can imagine that there are many successes in the comparison if you only have four possible residues like in nucleotide sequences. Therefore you can set a window size which is smoothing the dot plot. Instead of comparing single residues it compares subsequences of length set as window size. The score is now calculated with respect to aligning the subsequences.
- Threshold
The dot plot shows the calculated scores with colored threshold. Hence you can better recognize the most important similarities.

Examples and interpretations of dot plots

Contrary to simple sequence alignments dot plots can be a very useful tool for spotting various evolutionary events which may have happened to the sequences of interest.

Below is shown some examples of dot plots where sequence insertions, low complexity regions, inverted repeats etc. can be identified visually.

Similar sequences

The most simple example of a dot plot is obtained by plotting two homologous sequences of interest. If very similar or identical sequences are plotted against each other a diagonal line will occur.

The dot plot in figure 15.9 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>.

Repeated regions

Sequence repeats can also be identified using dot plots. A repeat region will typically show up as lines parallel to the diagonal line.

If the dot plot shows more than one diagonal in the same region of a sequence, the regions depending to the other sequence are repeated. In figure 15.11 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 15.12. In this figure, three frame shifts for the sequence on the y-axis are found.

1. Deletion of nucleotides

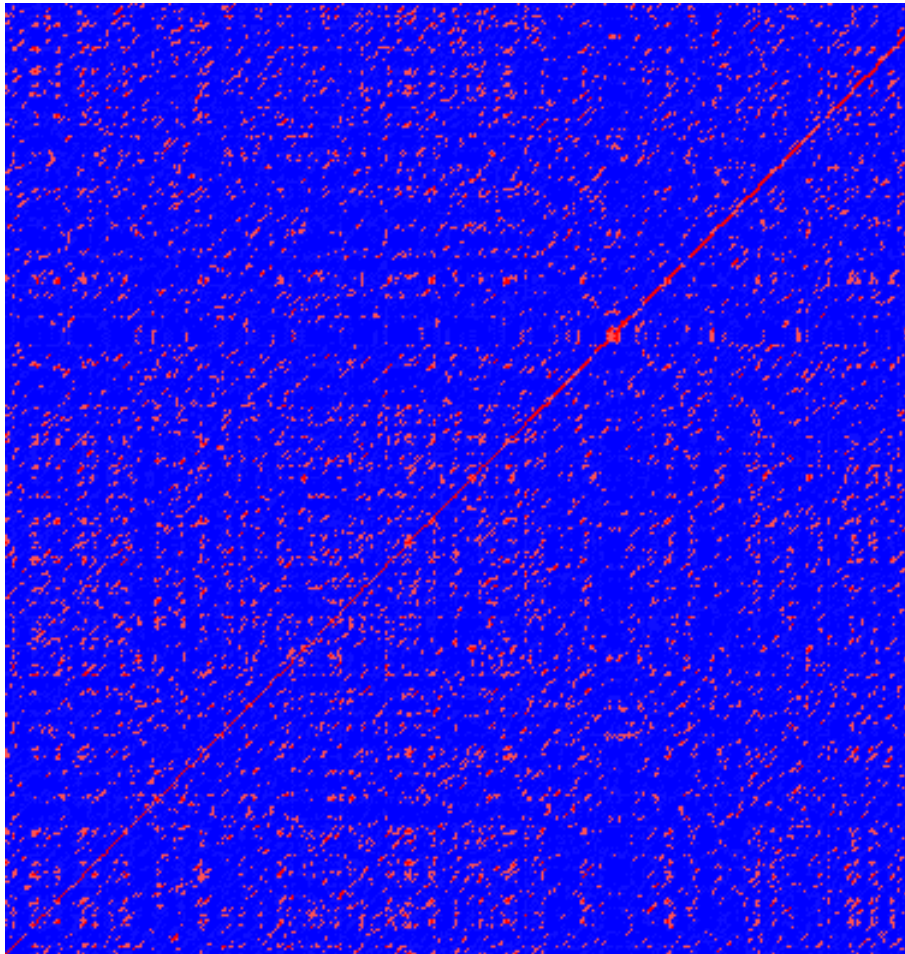


Figure 15.9: Dot plot of DQ232610 vs. DQ023146 (*Influenza A virus nucleoproteins*) showing and overall similarity

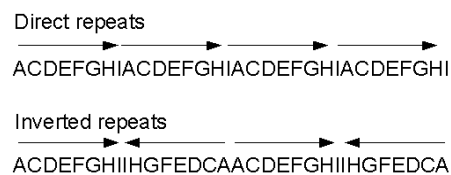


Figure 15.10: Direct and inverted repeats shown on an amino acid sequence generated for demonstration purposes.

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 15.13 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

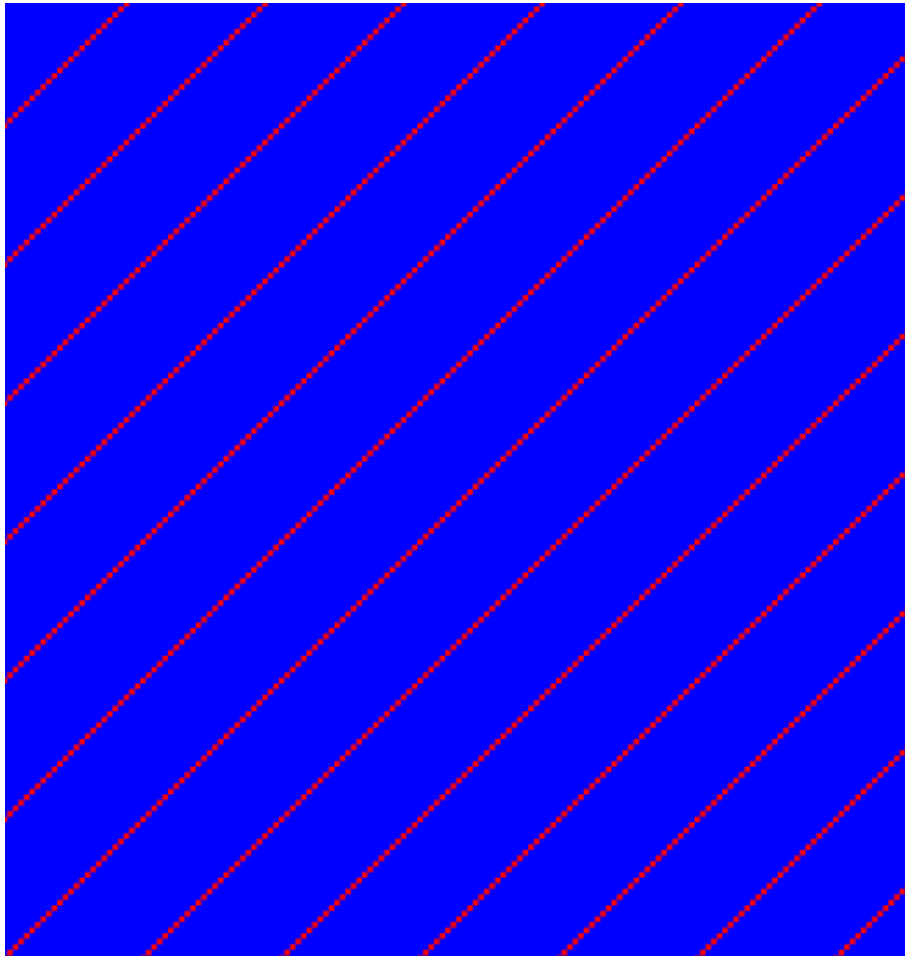


Figure 15.11: The dot plot of a sequence showing repeated elements. See also figure 15.10.

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 15.14 is a square shows the low-complexity region of this sequence.

15.4.4 Bioinformatics explained: Scoring matrices

Biological sequences have evolved throughout time and evolution has shown that not all changes to a biological sequence is equally likely to happen. Certain amino acid substitutions (change of one amino acid to another) happen often, whereas other substitutions are very rare. For instance, tryptophan (W) which is a relatively rare amino acid, will only – on very rare occasions – mutate into a leucine (L).

Based on evolution of proteins it became apparent that these changes or substitutions of amino acids can be modeled by a scoring matrix also referred to as a substitution matrix. See an example of a scoring matrix in table 15.1. 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

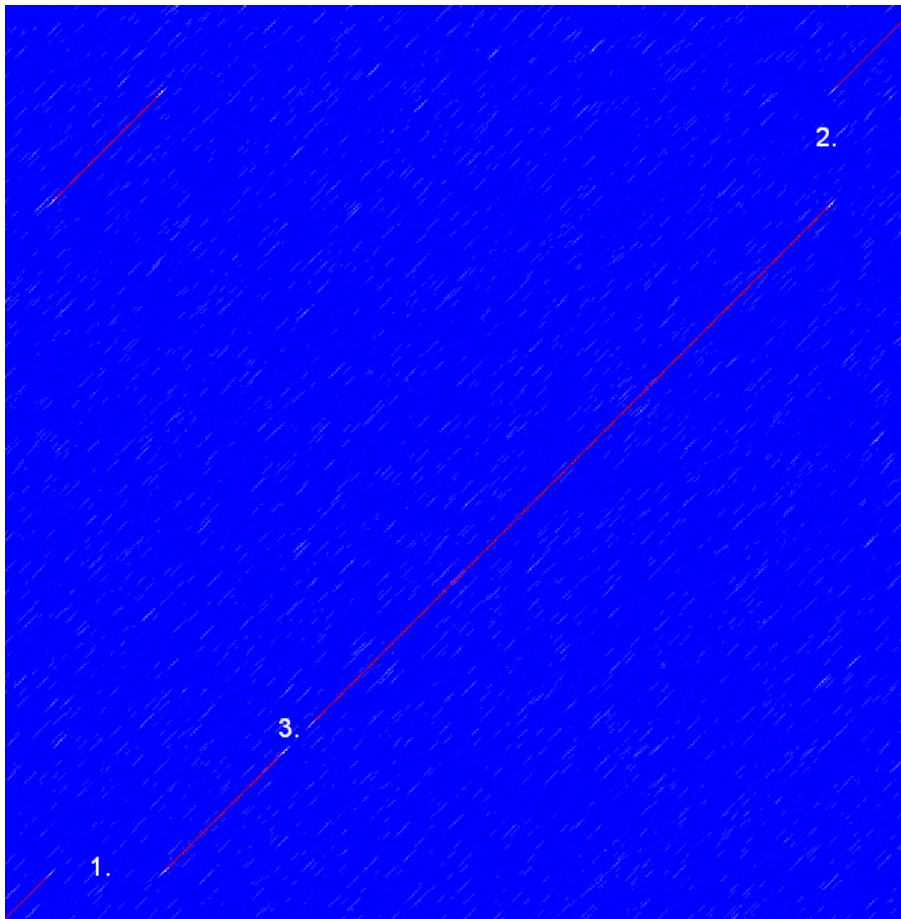


Figure 15.12: This dot plot show various frame shifts in the sequence. See text for details.

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 built 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 15.15).

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 built is very old by now, and the PAM matrices assume that all amino acids mutate at the same rate - this is not a correct assumption.

BLOSUM

In 1992, 14 years after the PAM matrices were published, the BLOSUM matrices (BLOCKS Substitution Matrix) were developed and published [Henikoff and Henikoff, 1992].

Henikoff et al. wanted to model more divergent proteins, thus they used locally aligned

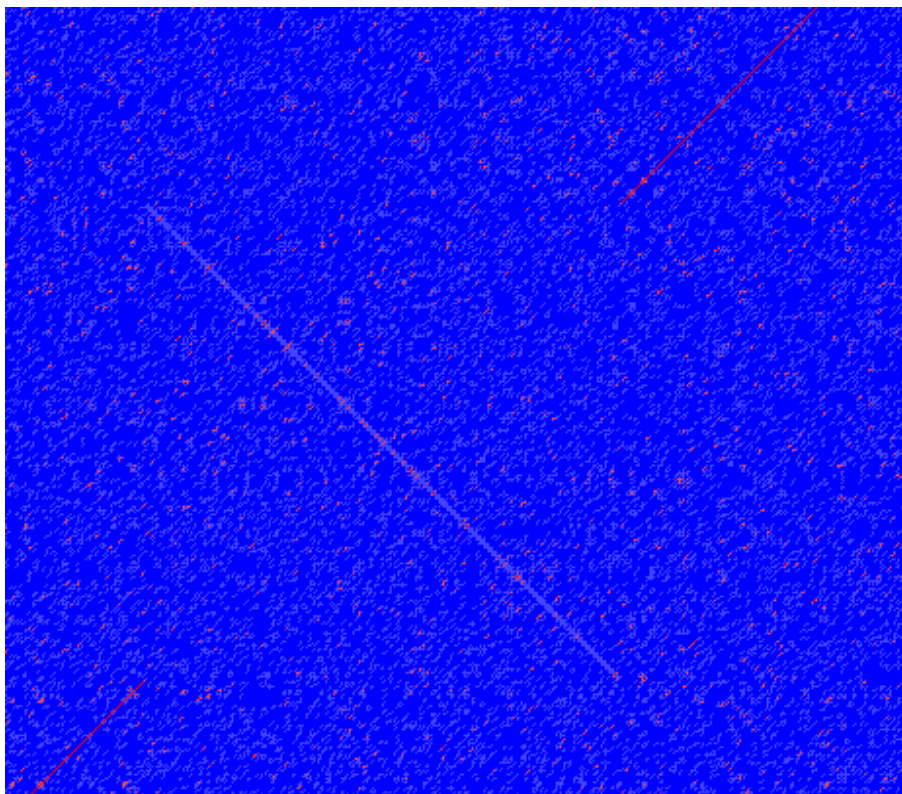


Figure 15.13: The dot plot showing an inversion in a sequence. See also figure 15.10.

sequences where none of the aligned sequences share less than 62% identity. This resulted in a scoring matrix $\frac{1}{2}$ 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 to 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 probably strongly influence on the outcome of the analysis. In general a few rules apply to the selection of scoring matrices.

- For closely related sequences choose BLOSUM matrices created for highly similar alignments, like BLOSUM80. You can also select low PAM matrices such as PAM1.
- For distant related sequences, select low BLOSUM matrices (for example BLOSUM45) or high PAM matrices such as PAM250.

The BLOSUM matrices with low numbers correspond to PAM matrices with high numbers. (See figure 15.15) for correlations between the PAM and BLOSUM matrices. To summarize, if you

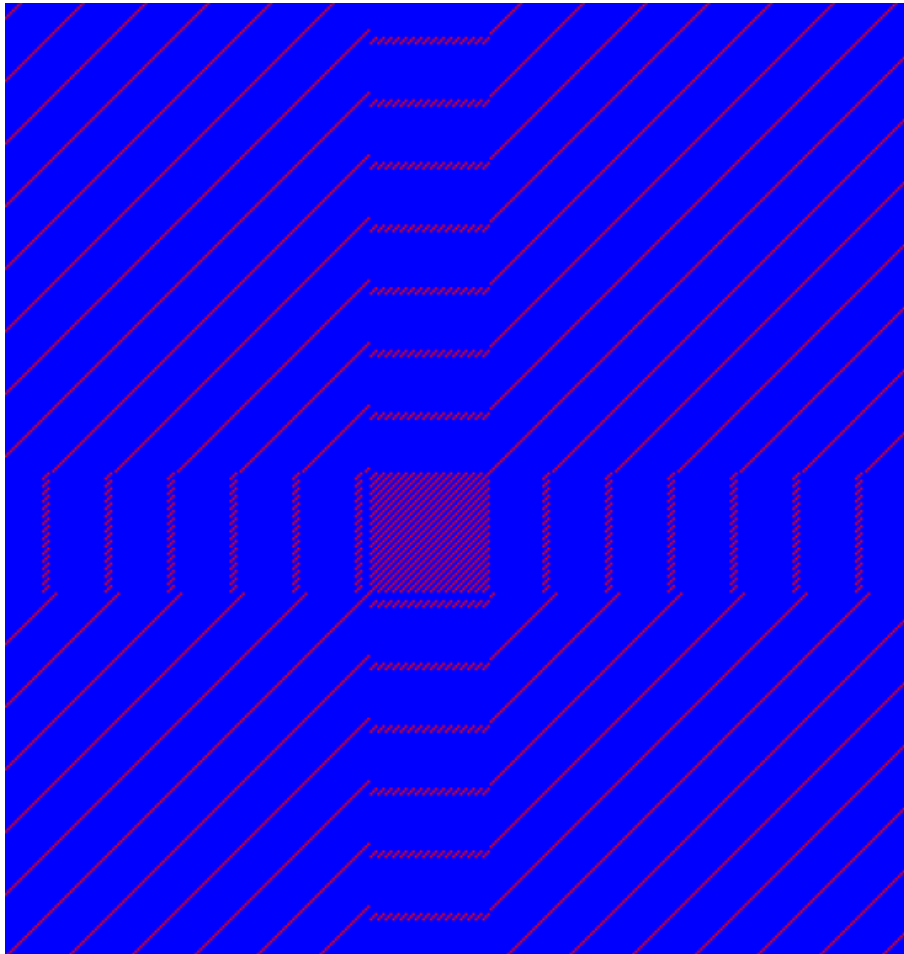


Figure 15.14: The dot plot showing a low-complexity region in the sequence. The sequence is artificial and low complexity regions do not always show as a square.

want to find distant related proteins to a sequence of interest using BLAST, you could benefit of using BLOSUM45 or similar matrices.

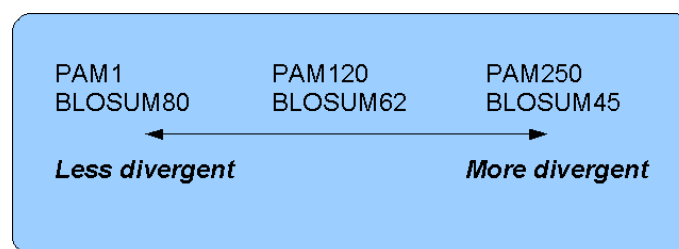


Figure 15.15: Relationship between scoring matrices. The BLOSUM62 has become a de facto standard scoring matrix for a wide range of alignment programs. It is the default matrix in BLAST.

Other useful resources

BLOKS database

<http://blocks.fhcrc.org/>

NCBI help site

http://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs

	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 15.1: **The BLOSUM62 matrix.** A tabular view of the BLOSUM62 matrix containing all possible substitution scores [Henikoff and Henikoff, 1992].

15.5 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:

Toolbox | General Sequence Analysis (🔧) | Create Complexity Plot (📊)

This opens a dialog. In **Step 1** you can use the arrows to change, remove and add DNA and protein sequences in the **Selected Elements** window.

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 15.16 shows an example of a local complexity plot.

Click **Finish** to start the tool. The values of the complexity plot approaches 1.0 as the distribution of amino acids become more complex.

See section A in the appendix for information about the graph view.

15.6 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:

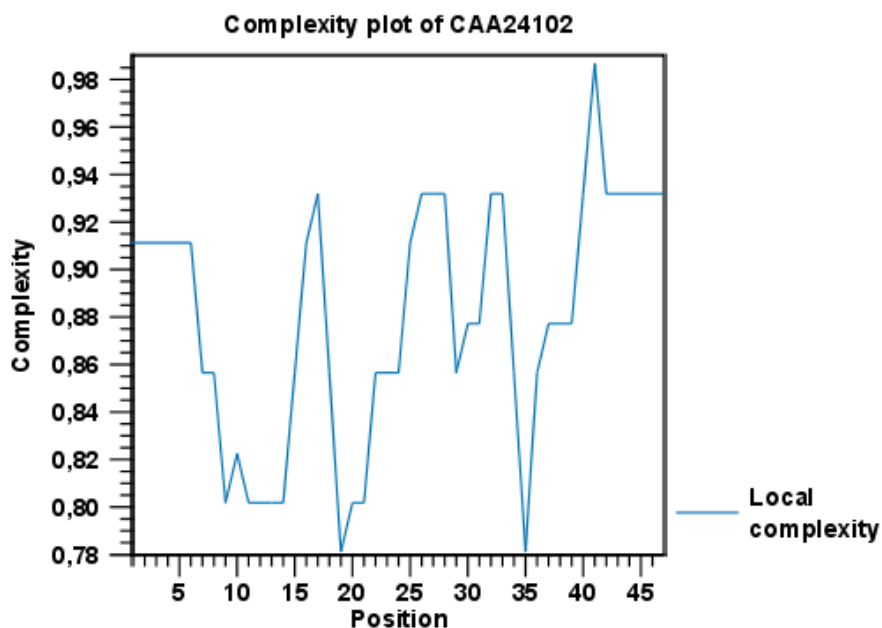


Figure 15.16: An example of a local complexity plot.

Toolbox | General Sequence Analysis (🔍) | Create Sequence Statistics (📊)

Select one or more sequence(s) or/and one or more sequence list(s). **Note!** You cannot create statistics for DNA and protein sequences at the same time, they must be run separately.

Next (figure 15.17), the dialog offers to adjust the following parameters:

- **Individual statistics layout.** If more sequences were selected in **Step 1**, this function generates separate statistics report for each sequence.
- **Comparative statistics layout.** If more sequences were selected in **Step 1**, this function generates statistics with comparisons between the sequences.

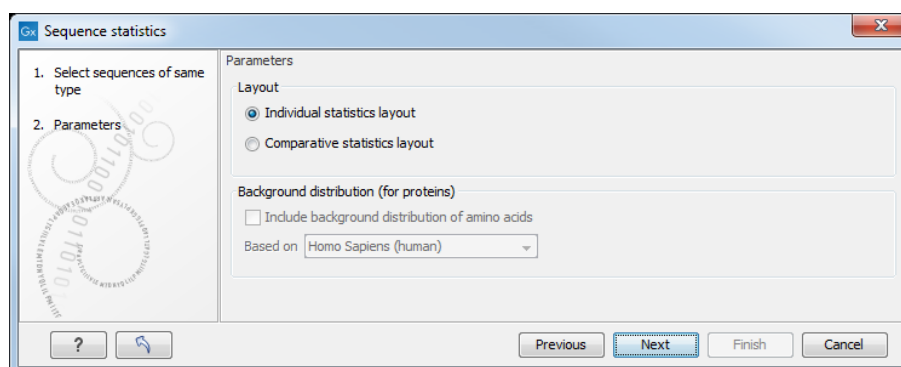


Figure 15.17: Setting parameters for the Sequence statistics tool.

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 **Finish** to start the tool. An example of protein sequence statistics is shown in figure 15.18.

1.1 Sequence information	
Sequence type	Protein
Length	147aa
Organism	Mus musculus
Name	HBB0_MOUSE
Description	RecName: Full=Hemoglobin subunit beta-H0; AltName: Full=Beta-H0-globin; AltName: Full=Hemoglobin beta-H0 chain
Modification Date	23-JAN-2007
Weight	16.384 kDa
Isoelectric point	9.08
Aliphatic index	95.578

Figure 15.18: 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 protein sequence statistics includes:

- **Sequence Information:**

- Sequence type
- Length
- Organism
- Name
- Description
- Modification Date
- Weight. This is calculated like this: $sum_{units\ in\ sequence}(weight(unit)) - links * weight(H_2O)$ 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

- Amino acid counts, frequencies
- Annotation counts

The output of nucleotide sequence statistics include:

- **General statistics:**
 - Sequence type
 - Length
 - Organism
 - Name
 - Description

- Modification Date
- Weight (calculated as single-stranded and double-stranded DNA)
- Annotation table
- Nucleotide distribution table

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 [15.6.1](#).

15.6.1 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.

$$\text{Aliphatic index} = X(\text{Ala}) + a * X(\text{Val}) + b * X(\text{Leu}) + b * (X)\text{Ile}$$

$X(\text{Ala})$, $X(\text{Val})$, $X(\text{Ile})$ and $X(\text{Leu})$ 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	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 15.2: **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*.

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 15.2). If leucine is found N-terminally in mammalian proteins the estimated half-life is 5.5 hours.

- **Extinction coefficient** This measure indicates how much light is absorbed by a protein at a particular wavelength. The extinction coefficient is measured by UV spectrophotometry, but can also be calculated. The amino acid composition is important when calculating the extinction coefficient. The extinction coefficient is calculated from the absorbance of cysteine, tyrosine and tryptophan using the following equation:

$$Ext(Protein) = count(Cystine) * Ext(Cystine) + count(Tyr) * Ext(Tyr) + count(Trp) * Ext(Trp)$$

where Ext is the extinction coefficient of amino acid in question. At 280nm the extinction coefficients are: Cys=120, Tyr=1280 and Trp=5690. This equation is only valid under the following conditions:

- pH 6.5
- 6.0 M guanidium hydrochloride
- 0.02 M phosphate buffer

The extinction coefficient values of the three important amino acids at different wavelengths are found in [Gill and von Hippel, 1989]. Knowing the extinction coefficient, the absorbance (optical density) can be calculated using the following formula: $Absorbance(Protein) = \frac{Ext(Protein)}{Molecular\ weight}$

Two values are reported. The first value is computed assuming that all cysteine residues appear as half cystines, meaning they form di-sulfide bridges to other cysteines. The second number assumes that no di-sulfide bonds are formed.

- **Atomic composition** Amino acids are indeed very simple compounds. All 20 amino acids consist of combinations of only five different atoms. The atoms which can be found in these simple structures are: Carbon, Nitrogen, Hydrogen, Sulfur, Oxygen. The atomic composition of a protein can for example be used to calculate the precise molecular weight of the entire protein.
- **Total number of negatively charged residues (Asp + Glu)** At neutral pH, the fraction of negatively charged residues provides information about the location of the protein. Intracellular proteins tend to have a higher fraction of negatively charged residues than extracellular proteins.
- **Total number of positively charged residues (Arg + Lys)** At neutral pH, nuclear proteins have a high relative percentage of positively charged amino acids. Nuclear proteins often bind to the negatively charged DNA, which may regulate gene expression or help to fold the DNA. Nuclear proteins often have a low percentage of aromatic residues [Andrade et al., 1998].
- **Amino acid distribution** Amino acids are the basic components of proteins. The amino acid distribution in a protein is simply the percentage of the different amino acids represented in a particular protein of interest. Amino acid composition is generally conserved through family-classes in different organisms which can be useful when studying a particular protein or enzymes across species borders. Another interesting observation is that amino acid composition variate slightly between proteins from different subcellular localizations. This fact has been used in several computational methods, used for prediction of subcellular localization.
- **Annotation table** This table provides an overview of all the different annotations associated with the sequence and their incidence.
- **Dipeptide distribution** This measure is simply a count, or frequency, of all the observed adjacent pairs of amino acids (dipeptides) found in the protein. It is only possible to report neighboring amino acids. Knowledge on dipeptide composition have previously been used for prediction of subcellular localization.

15.7 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:

Toolbox | General Sequence Analyses | Join sequences 

This opens the dialog shown in figure 15.19.

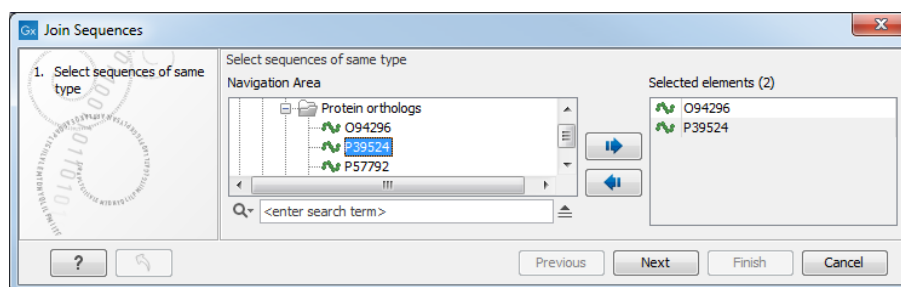


Figure 15.19: Selecting two sequences to be joined.

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 15.20.

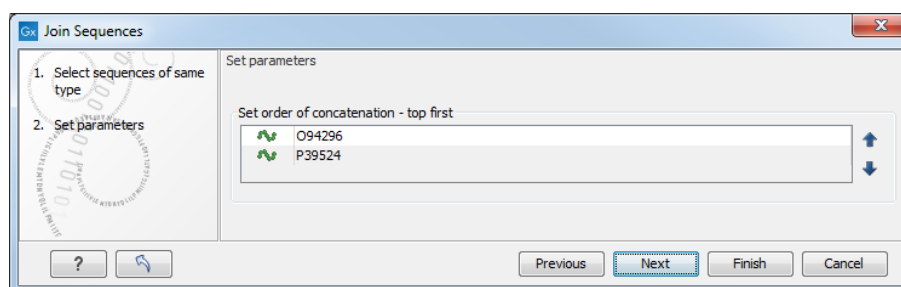


Figure 15.20: Setting the order in which sequences are joined.

In step 2 you can change the order in which the sequences will be joined. Select a sequence and use the arrows to move the selected sequence up or down.

Click **Finish** to start the tool.

The result is shown in figure 15.21.

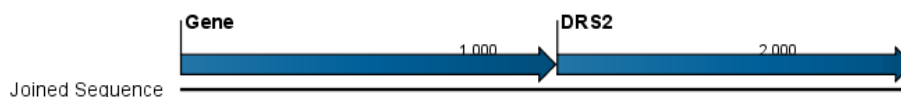


Figure 15.21: The result of joining sequences is a new sequence containing the annotations of the joined sequences (they each had a HBB annotation).

15.8 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:

Toolbox | General Sequence Analysis (🔍) | Pattern Discovery (🔍?)

Choose one or more sequence(s) or sequence list(s). 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 15.22).

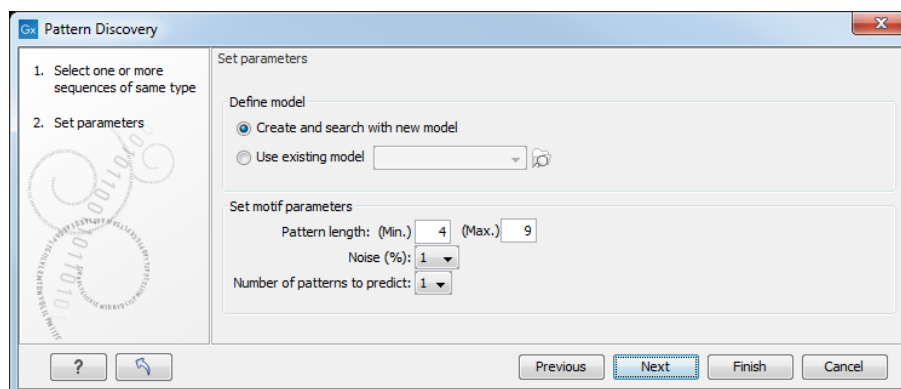


Figure 15.22: Setting parameters for the pattern discovery. See text for details.

In order to search unknown sequences with an already existing model:

Select to use an already existing model which is seen in figure 15.22. Models are represented with the following icon in the **Navigation Area** (🔍).

15.8.1 Pattern discovery search parameters

Various parameters can be set prior to the pattern discovery. The parameters are listed below and a screenshot of the parameter settings can be seen in figure 15.22.

- **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 **Finish** to start the tool. This will open a view showing the patterns found as annotations on the original sequence (see figure 15.23). If you have selected several sequences, a corresponding number of views will be opened.



Pattern1
3VCNKNQTA EDLAWSYGFP ECARFLTMIK CMQTARSSGE
Pattern1

Figure 15.23: Sequence view displaying two discovered patterns.

15.8.2 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.

15.9 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 20.1.1). This approach is called *Dynamic motifs* and is an easy way to spot known sequence motifs when working with sequences for cloning etc.
- A 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.

15.9.1 Dynamic motifs

In the **Side Panel** of sequence views, there is a group called **Motifs** (see figure 15.24).

The Workbench will look for the listed motifs in the sequence that is open and by clicking the check box next to the motif it will be shown in the view as illustrated in figure 15.25.



Figure 15.24: Dynamic motifs in the Side Panel.

```

      380           400           420
      |           |           |
      CCCATTGACGTCAATGGGAGTTTGTGTTTGGCACCAAATCAACGGGACTTTCC
      |           |           |
      440           460
      |           |
      AAAATGTCGTAACAACCTCCGCCCCATTGACGCAAAATGGGCGGTAGGCGTGTAC
      |           |           |
      480           500           520
      |           |           |
      GGTGGGAGGTCTATATAAGCAGAGCTCGTTTAGTGAACCGTCAAGATCGCCTGG
  
```

Figure 15.25: Showing dynamic motifs on the sequence.

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 15.26.

```

      3CCCATTGACGCAAAATGGGCGGTAGGCGTGTACCGTGGGAGG
      |
      /motif=CGCAAATGGGCGGTAGGCGTG, list index: 5 (CMV);
      /type=Simple
      /description=CMV promoter primer
  
```

Figure 15.26: 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 15.24 is a pre-defined list that is included with the workbench, but you can define your own set of motifs to use instead. In order to do this, you can either launch the Create Motif List tool from the Navigation Area or using the **Add Motif** button in the side panel (see section 15.10)). Once your list of custom motif(s) is saved, you can click the **Manage Motifs** button in the side panel which will bring up the dialog shown in figure 15.27.

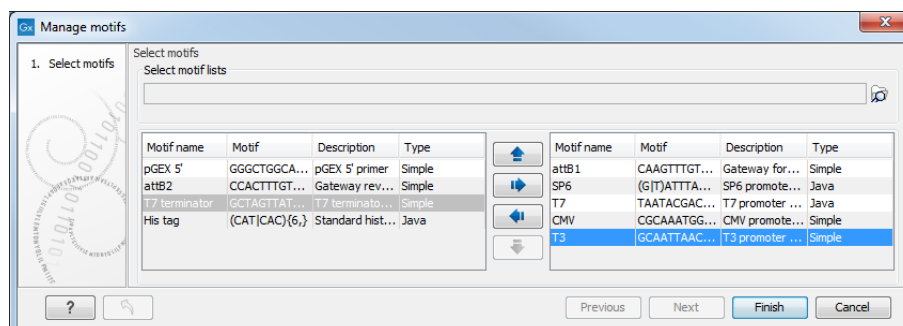



Figure 15.27: Managing the motifs to be shown.

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**.

15.9.2 Motif search from the Toolbox

The dynamic motifs described in section 15.9.1 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, go to:

Toolbox | General Sequence Analysis  | **Motif Search** 

A dialog window will be launched. Use the arrows to add or remove sequences or sequence lists between the Navigation Area and the selected elements.

You can perform the analysis on several DNA or several protein sequences at a time. In this case, 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 15.28).

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 15.9.3.

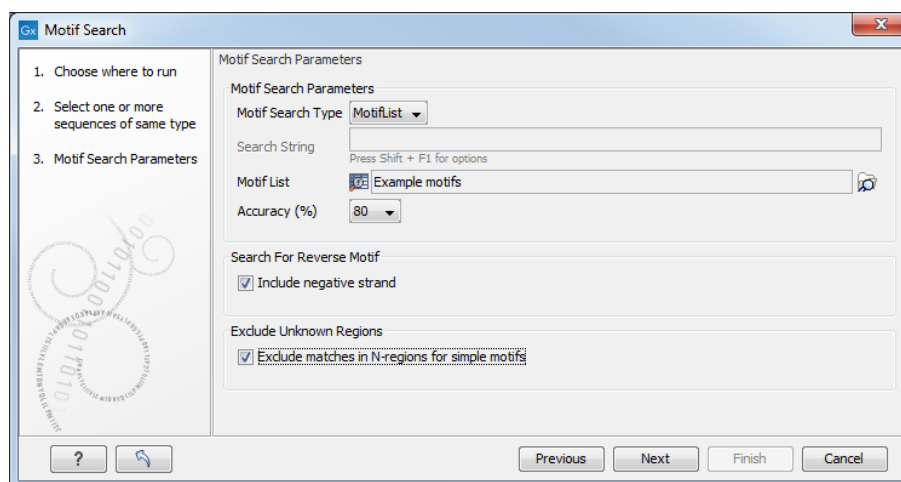


Figure 15.28: Setting parameters for the motif search.

- 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 15.10).
- **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 15.9.3. 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 15.29).
- **Create table.** This will create an overview table of all the motifs found for all the input sequences.

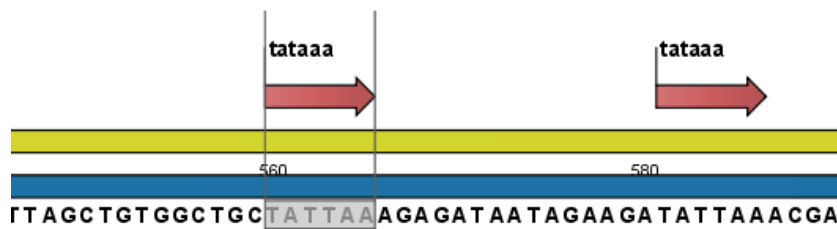


Figure 15.29: Sequence view displaying the pattern found. The search string was 'tataaa'.

15.9.3 Java regular expressions

A regular expressions is a string that describes or matches a set of strings, according to certain syntax rules. They are usually used to give a concise description of a set, without having to list all elements. The simplest form of a regular expression is a literal string. The syntax used for the regular expressions is the Java regular expression syntax (see <http://java.sun.com/docs/books/tutorial/essential/regex/index.html>). Below is listed some of the most important syntax rules which are also shown in the help pop-up when you press Shift + F1:

`[A-Z]` will match the characters A through Z (Range). You can also put single characters between the brackets: The expression `[AGT]` matches the characters A, G or T.

`[A-D[M-P]]` will match the characters A through D and M through P (Union). You can also put single characters between the brackets: The expression `[AG[M-P]]` matches the characters A, G and M through P.

`[A-M&&[H-P]]` will match the characters between A and M lying between H and P (Intersection). You can also put single characters between the brackets. The expression `[A-M&&[HGTDA]]` matches the characters A through M which is H, G, T, D or A.

`[^A-M]` will match any character except those between A and M (Excluding). You can also put single characters between the brackets: The expression `[^AG]` matches any character except A and G.

`[A-Z&&[^M-P]]` will match any character A through Z except those between M and P (Subtraction). You can also put single characters between the brackets: The expression `[A-P&&[^CG]]` matches any character between A and P except C and G.

The symbol `.` matches any character.

`X{n}` will match a repetition of an element indicated by following that element with a numerical value or a numerical range between the curly brackets. For example, `ACG{2}` matches the string `ACGG` and `(ACG){2}` matches `ACGACG`.

$X\{n,m\}$ will match a certain number of repetitions of an element indicated by following that element with two numerical values between the curly brackets. The first number is a lower limit on the number of repetitions and the second number is an upper limit on the number of repetitions. For example, $ACT\{1,3\}$ matches ACT , $ACTT$ and $ACTTT$.

$X\{n,\}$ represents a repetition of an element at least n times. For example, $(AC)\{2,\}$ matches all strings $ACAC$, $ACACAC$, $ACACACAC$,...

The symbol \wedge restricts the search to the beginning of your sequence. For example, if you search through a sequence with the regular expression $\wedge AC$, 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 $[ACG][\wedge AC]G\{2\}$ matches all strings of length 4, where the first character is A,C or G and the second is any character except A,C and the third and fourth character is G . The expression $G.[\wedge A]\$$ matches all strings of length 3 in the end of your sequence, where the first character is C , the second any character and the third any character except A .

15.10 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 can be created using:

Toolbox | General Sequence Analysis (📁) | Create Motif List (🛠️)

Click on the **Add (+)** button at the bottom of the view. This will open a dialog shown in figure 15.30.

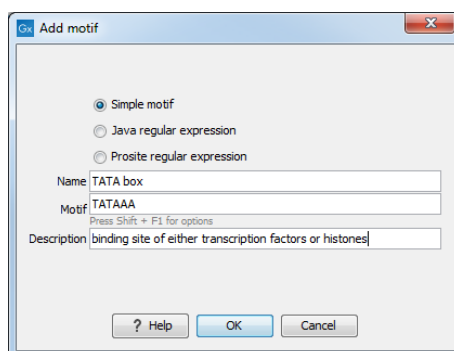




Figure 15.30: Entering a new motif in the list.


In this dialog, you can enter the following information:


- **Name.** The name of the motif. In the result of a motif search, this name will appear as the name of the annotation and in the result table.
- **Motif.** The actual motif. See section 15.9.2 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 will be 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 [15.9.2](#).

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". Note that reformatting Prosite file into FASTA format for import will fail, as only simple motifs can be imported this way and regular expressions are not supported.

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 [15.9](#)).

Chapter 16

Nucleotide analyses

Contents

16.1 Convert DNA to RNA	319
16.2 Convert RNA to DNA	320
16.3 Reverse complements of sequences	320
16.4 Reverse sequence	321
16.5 Translation of DNA or RNA to protein	322
16.6 Find open reading frames	323
16.6.1 Open reading frame parameters	324

CLC Main Workbench offers different kinds of sequence analyses, which only apply to DNA and RNA.

16.1 Convert DNA to RNA

CLC Main Workbench lets you convert a DNA sequence into RNA, substituting the T residues (Thymine) for U residues (Uracil):

Toolbox | Nucleotide Analysis (📄) | Convert DNA to RNA (🔄)

This opens the dialog displayed in figure 16.1:

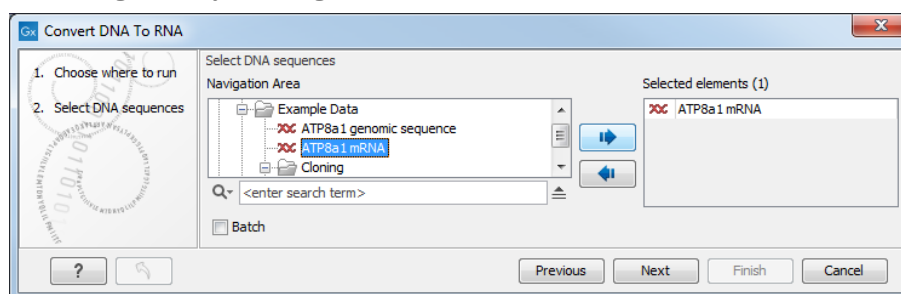


Figure 16.1: Translating DNA to RNA.

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 **Finish** to start the tool.

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.

16.2 Convert RNA to DNA

CLC Main Workbench lets you convert an RNA sequence into DNA, substituting the U residues (Uracil) for T residues (Thymine):

Toolbox | Nucleotide Analysis (📄) | Convert RNA to DNA (🔄)

This opens the dialog displayed in figure 16.2:

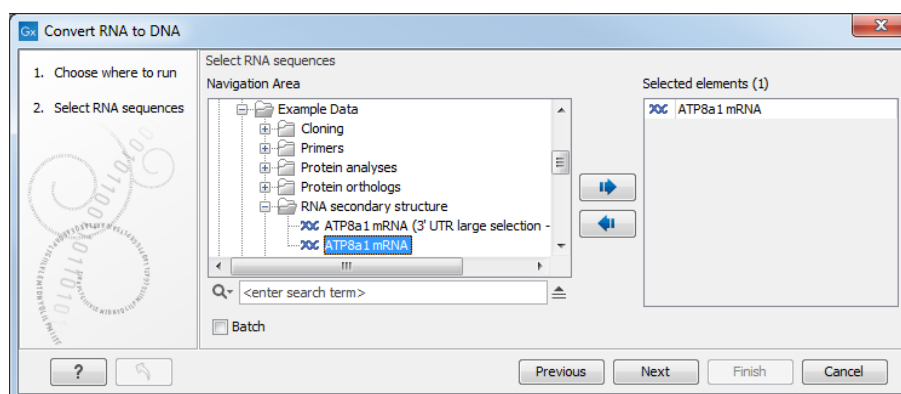


Figure 16.2: *Translating RNA to DNA.*

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 **Finish** to start the tool.

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.

16.3 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:

Toolbox | Nucleotide Analysis (📄)| Reverse Complement Sequence (🔄)

This opens the dialog displayed in figure 16.3:

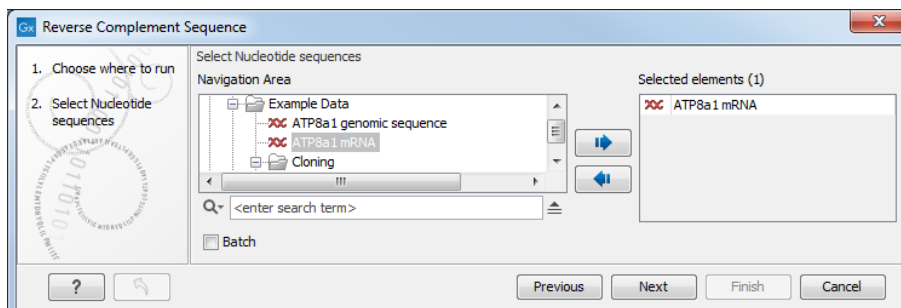


Figure 16.3: Creating a reverse complement sequence.

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 **Finish** to start the tool.

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.

16.4 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 16.3.

To run the tool, go to:

Toolbox | Nucleotide Analysis (📄)| Reverse Sequence (🔄)

This opens the dialog displayed in figure 16.4:

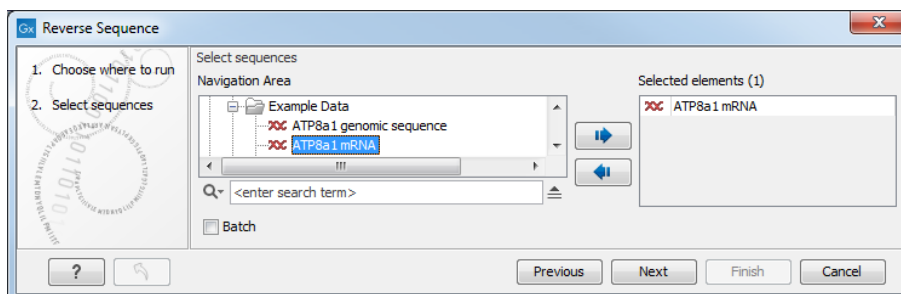


Figure 16.4: Reversing a sequence.

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 **Finish** to start the tool.

The Reverse Sequence tool will output reversed sequences holding the input sequences name with a "-R" suffix.

Note! This is not the same as a reverse complement. If you wish to create the reverse complement, please refer to section 16.3.

16.5 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, go to:

Toolbox | Nucleotide Analysis (📄) | Translate to Protein (🧬)

This opens the dialog displayed in figure 16.5:

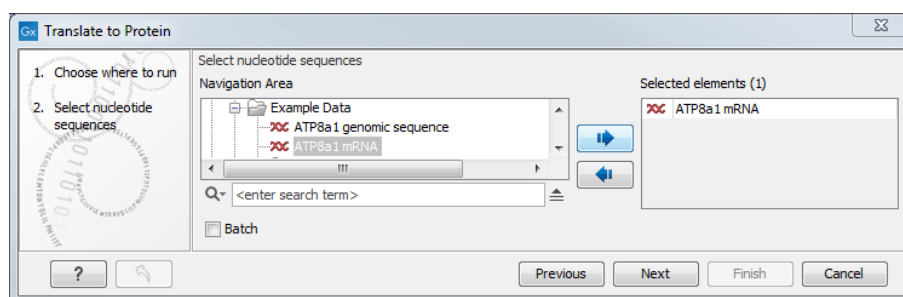


Figure 16.5: Choosing sequences for translation.

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Clicking **Next** generates the dialog seen in figure 16.6:

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 CDS 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. The "Extract existing translations from annotation" allows to list the amino acid CDS sequence shown in the tool tip annotation (e.g. interstate from NCBI download) and does therefore not represent a translation of the actual nt 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

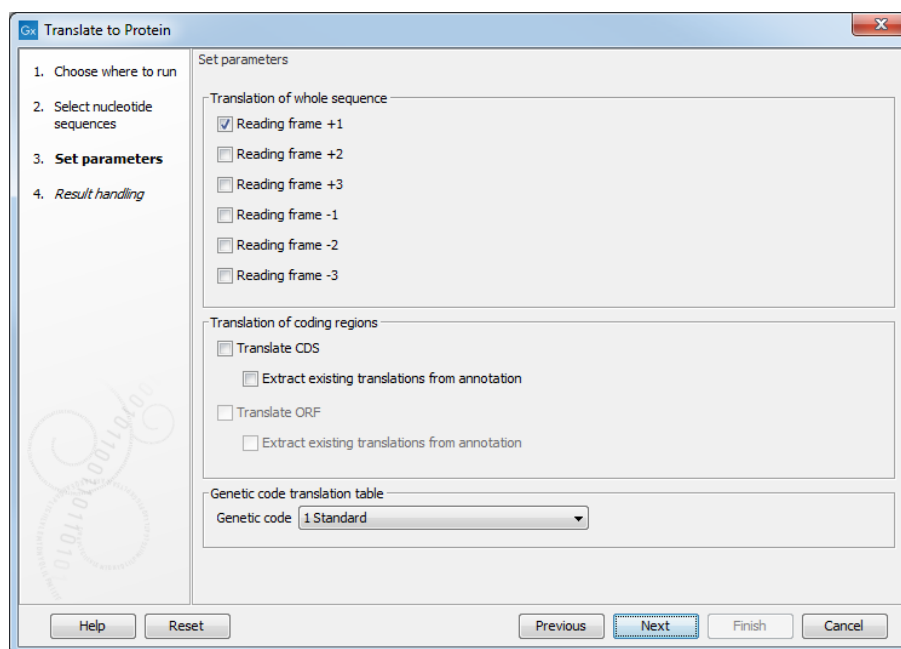


Figure 16.6: Choosing translation of CDSs using standard translation table.

printable version of the user manual. Instead, the tables are included in the **Help**-menu in the **Menu Bar** (in the appendix).

Click **Finish** to start the tool. 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.

The name for a coding region translation consists of the name of the input sequence followed by the annotation type and finally the annotation name.

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/ORF..." in the translation dialog (see figure 16.6).

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... (👉)

A dialog opens to offer you the following choices (figure 16.7): either a specific genetic code translation table, or to extract the existing translation from annotation (if the annotation contains information about the translation). Choose the option needed and click **Translate**.

The CDS and ORF annotations are colored yellow as default.

16.6 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

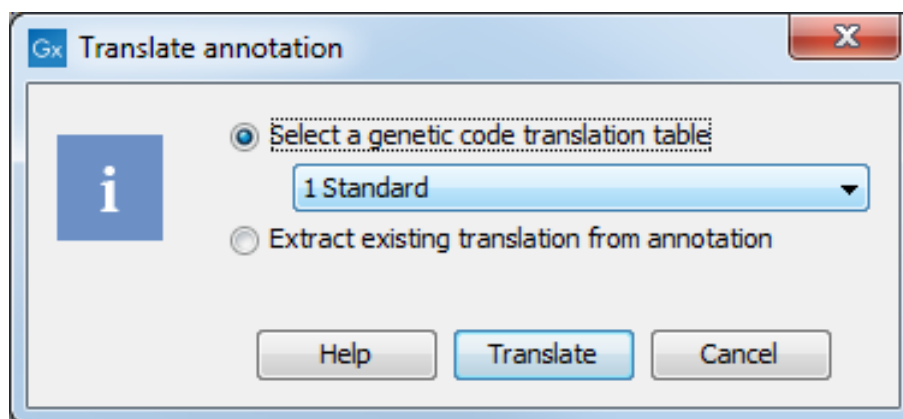


Figure 16.7: Choosing how to translate CDS or ORF annotations.

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:

Toolbox | Nucleotide Analysis (📄) | Find Open Reading Frames (🔍)

This opens the dialog displayed in figure 16.8:

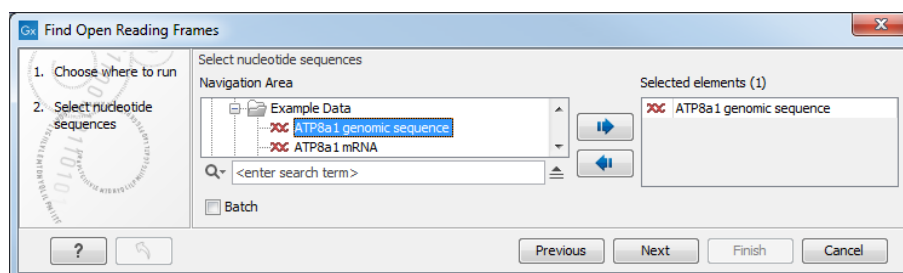


Figure 16.8: Create Reading Frame dialog.

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.

The **Find Open Reading Frames** tool simply looks for start and stop codons and reports any open reading frames that satisfy the parameters. If you want to adjust the parameters for finding open reading frames click **Next**.

16.6.1 Open reading frame parameters

This opens the dialog displayed in figure 16.9:

The adjustable parameters for the search are:

- **Start codon:**
 - **AUG**. Most commonly used start codon.

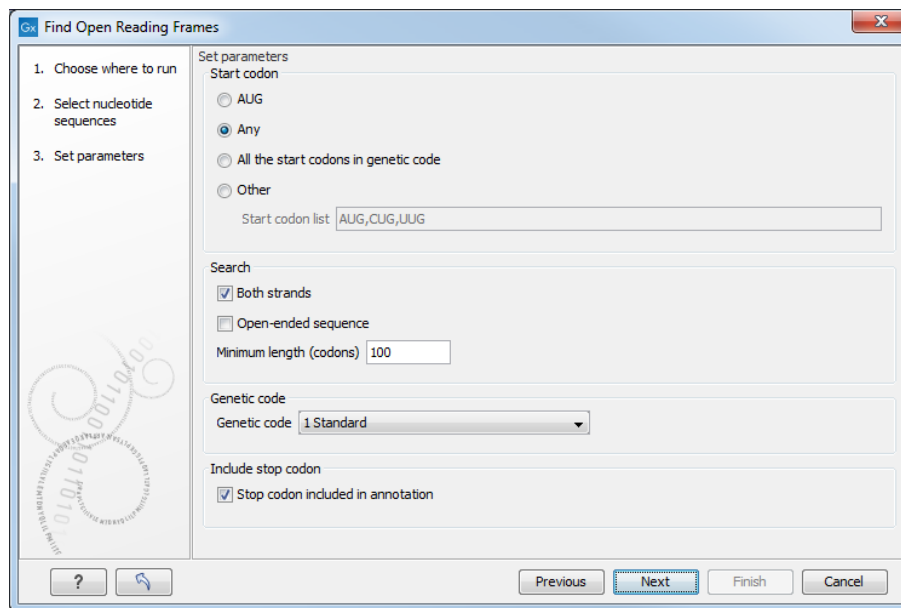


Figure 16.9: Parameters for the Reading Frame tool.

- **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 16.10).

Click **Finish** to start the tool.

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.

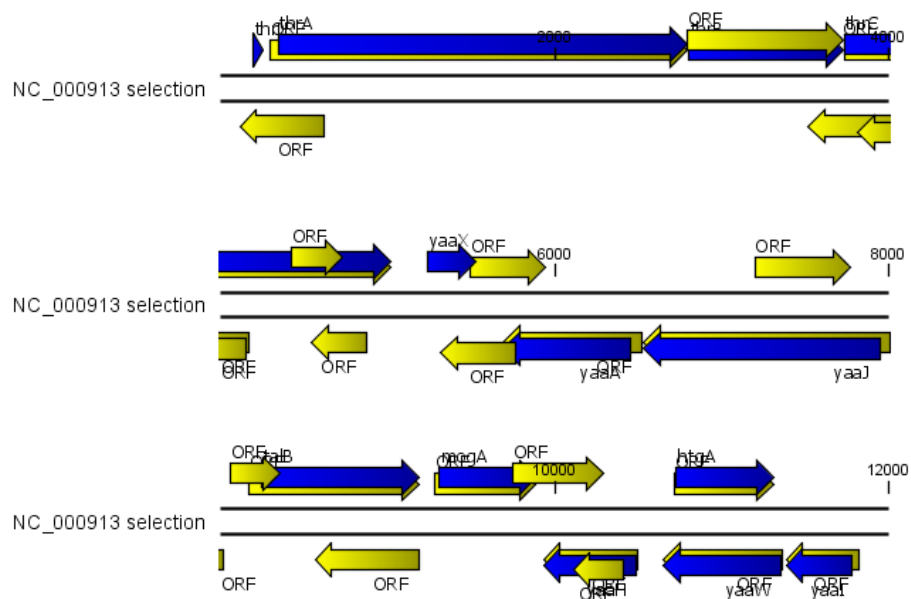


Figure 16.10: The first 12,000 positions of the *E. coli* sequence NC_000913 downloaded from GenBank. The blue (dark) annotations are the genes while the yellow (brighter) annotations are the ORFs with a length of at least 100 amino acids. On the positive strand around position 11,000, a gene starts 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.

Chapter 17

Protein analyses

Contents

17.1 Protein charge	327
17.2 Antigenicity	329
17.3 Hydrophobicity	330
17.3.1 Hydrophobicity graphs along sequence	331
17.3.2 Bioinformatics explained: Protein hydrophobicity	332
17.4 Pfam domain search	334
17.4.1 Download of Pfam database	335
17.4.2 Running Pfam Domain Search	335
17.5 Secondary structure prediction	337
17.6 Protein report	338
17.7 Reverse translation from protein into DNA	340
17.7.1 Bioinformatics explained: Reverse translation	342
17.8 Proteolytic cleavage detection	343
17.8.1 Bioinformatics explained: Proteolytic cleavage	345

CLC Main Workbench offers a number of analyses of proteins as described in this chapter.

Note that the SignalP plugin allows you to predict signal peptides. For more information, please read the plugin manual at http://resources.qiagenbioinformatics.com/manuals/signalP/current/SignalP_User_Manual.pdf.

The TMHMM plugin allows you to predict transmembrane helix. For more information, please read the plugin manual at http://resources.qiagenbioinformatics.com/manuals/tmhmm/current/Tmhmm_User_Manual.pdf.

17.1 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:

Toolbox | Protein Analysis (🌿) | Create Protein Charge Plot (📊)

This opens the dialog displayed in figure 17.1:

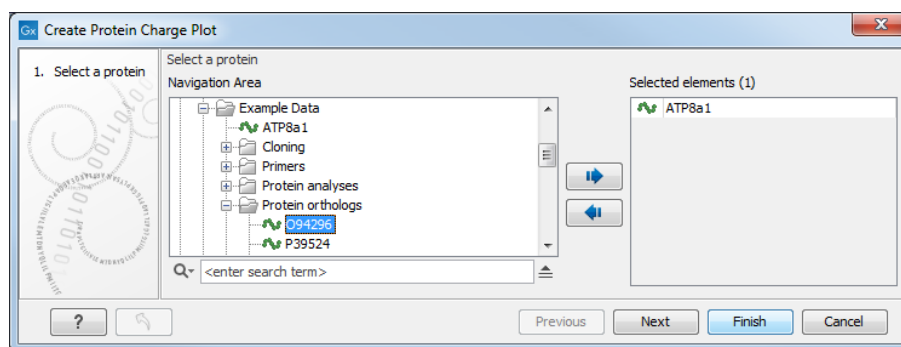


Figure 17.1: Choosing protein sequences to calculate protein charge.

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 **Finish** to start the tool.

Figure 17.2 shows the electrical charges for three proteins. In the **Side Panel** to the right, you can modify the layout of the graph.

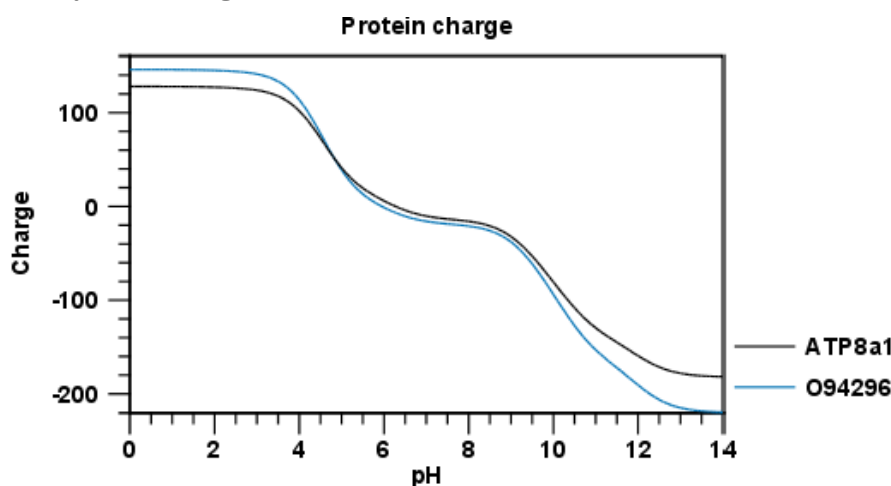


Figure 17.2: View of the protein charge.

See section A in the appendix for information about the graph view.

17.2 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 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 methods can not always be expected as the two methods are based on different training sets.

Displaying the antigenicity for a protein sequence in a plot is done in the following way:

Toolbox | Protein Analysis (📁) | Create Antigenicity Plot (📊)

This opens a dialog. The first step allows you to add or remove sequences. If you had already selected sequences in the Navigation Area before running the Toolbox action, these are shown in the **Selected Elements**. Clicking **Next** takes you through to **Step 2**, which is displayed in figure 17.3.

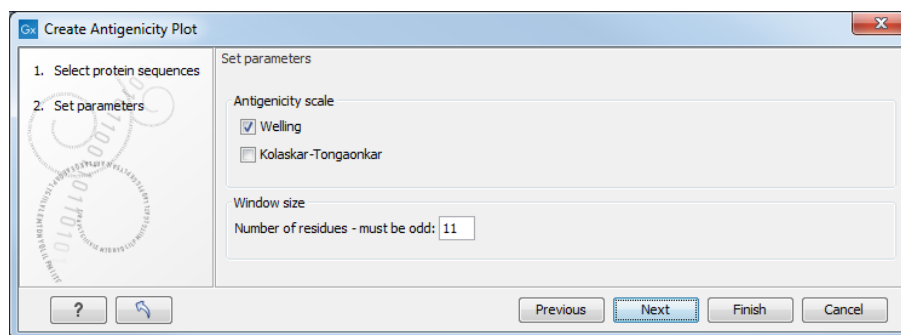


Figure 17.3: Step two in the Antigenicity Plot allows you to choose different antigenicity scales and the window size.

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 choose from a number of antigenicity scales. Click **Finish** to start the tool. The result can be seen in figure 17.4.

See section A 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 sum of the values in a 'window', which is a particular range of the sequence. The window length

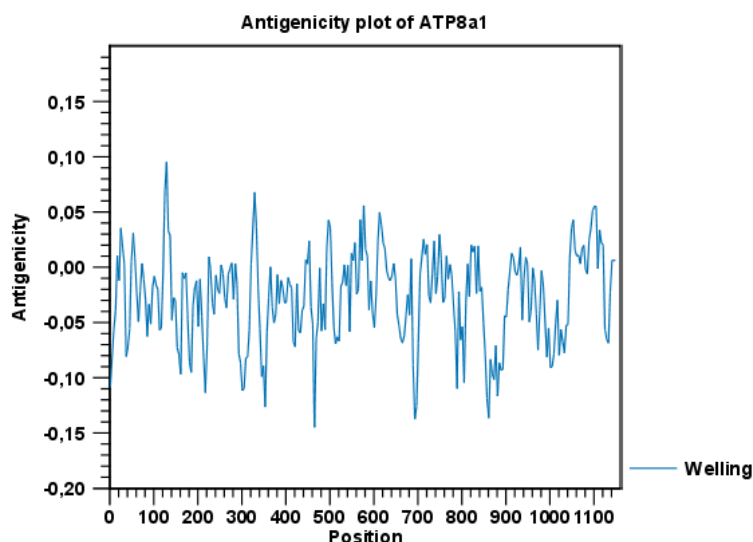


Figure 17.4: The result of the antigenicity plot calculation and the associated Side Panel.

can be set from 5 to 25 residues. The wider the window, the less fluctuations in the antigenicity scores.

Antigenicity graphs along the sequence can be displayed using the **Side Panel**. The functionality is similar to hydrophobicity (see section 17.3.1).

17.3 Hydrophobicity

CLC Main Workbench can calculate the hydrophobicity of protein sequences in different ways, using different algorithms. (See section 17.3.2). 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.

Displaying the hydrophobicity for a protein sequence in a plot is done in the following way:

Toolbox | Protein Analysis (📁) | **Create Hydrophobicity Plot** (📊)

This opens a dialog. The first step allows you to add or remove sequences. If you had already selected a sequence in the Navigation Area, this will be shown in the **Selected Elements**. Clicking **Next** takes you through to **Step 2**, which is displayed in figure 17.5.

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 choose from a number of hydrophobicity scales which are further explained in section 17.3.2. Click **Finish** to start the tool. The result can be seen in figure 17.6.

See section A in the appendix for information about the graph view.

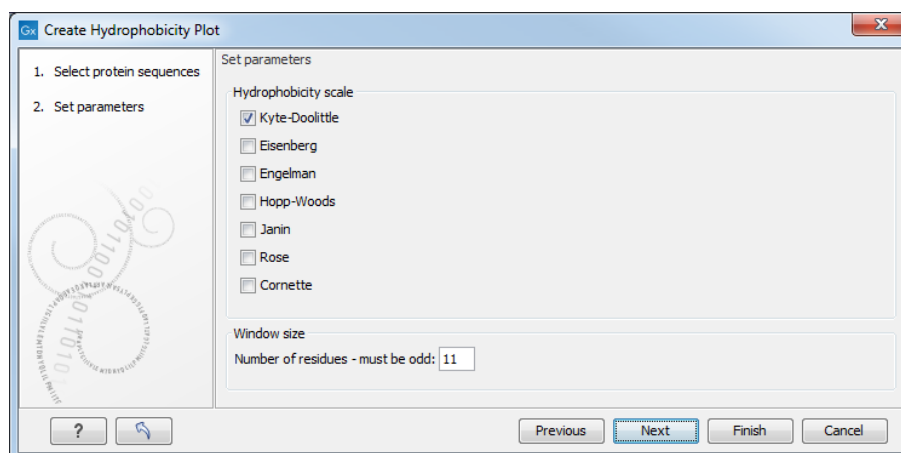


Figure 17.5: Step two in the *Hydrophobicity Plot* allows you to choose hydrophobicity scale and the window size.

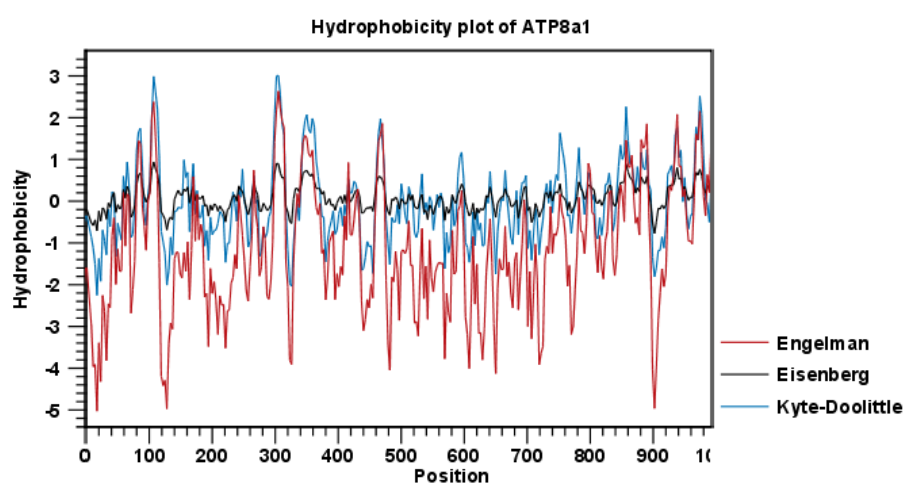


Figure 17.6: The result of the hydrophobicity plot calculation and the associated Side Panel.

17.3.1 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 17.7.

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 17.3.2).

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



Figure 17.7: The different available scales in Protein info in **CLC Main Workbench**.

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 17.8).



Figure 17.8: 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 17.8. Notice that you can choose the height of the graphs underneath the sequence.

17.3.2 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.

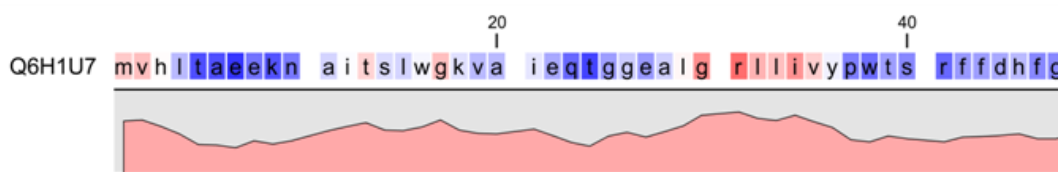


Figure 17.9: Plot of hydrophobicity along the amino acid sequence. Hydrophobic regions on the sequence have higher numbers according to the graph below the sequence, furthermore hydrophobic regions are colored on the sequence. Red indicates regions with high hydrophobicity and blue indicates regions 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 17.9).

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

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 17.1: *Hydrophobicity scales. This table shows seven different hydrophobicity scales which are generally used for prediction of e.g. transmembrane regions and antigenicity.*

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.

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>

17.4 Pfam domain search

With *CLC Main Workbench* you can perform a search for domains in protein sequences using the Pfam database. The Pfam database [Bateman et al., 2004] at <http://pfam.sanger.ac.uk/> was initially developed to aid the annotation of the *C. elegans* genome. The database is a large

collection of multiple sequence alignments that cover 14831 protein domains and protein families as of March 2014. The database contains profile hidden Markov models (HMMs) for individual domain alignments, which can be used to quickly identify domains in protein sequences.

Many proteins have a unique combination of domains, which can be responsible for e.g. the catalytic activities of enzymes. Annotating 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]. For example, a protein may be annotated incorrectly as an enzyme if the pairwise alignment only finds a regulatory domain.

Using the **Pfam Domain Search** tool in *CLC Main Workbench*, you can search for domains in sequence data which otherwise do not carry any annotation information. The domain search is performed using the `hmmsearch` tool from the HMMER3 package version 3.1b1 (<http://hmmer.janelia.org/>). The Pfam search tool annotates protein sequences with all domains in the Pfam database that have a significant match. It is possible to lower the significance cutoff thresholds in the `hmmsearch` algorithm, which will reduce the number of domain annotations. Individual domain annotations can be removed manually as described in section 11.3.4.

17.4.1 Download of Pfam database

To be able to run the **Pfam Domain Search** tool you must first download the Pfam database. The Pfam database can be downloaded using:

Toolbox | Protein Analysis  | **Download Pfam Database** 

Specify where you would like to save the downloaded Pfam database. The output of the **Download Pfam Database** tool is a database object, which can be selected as a parameter for the Pfam Domain Search tool. It doesn't really make sense to try to open the database object directly from the **Navigation Area** as all you can see directly is the element history (which version of the Workbench that has been used and the name of the downloaded files) and the element info, which in this case only provides information about the database name.

17.4.2 Running Pfam Domain Search

When you have downloaded the Pfam database you are ready to perform a Pfam domain search. To do this start the Pfam search tool:

Toolbox | Protein Analysis  | **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. Click **Next** to adjust parameters (see figure 17.10).

- **Database.** Choose which database to use when searching for Pfam domains. For information on how to download a Pfam database see section 17.4.1
- Significance cutoff

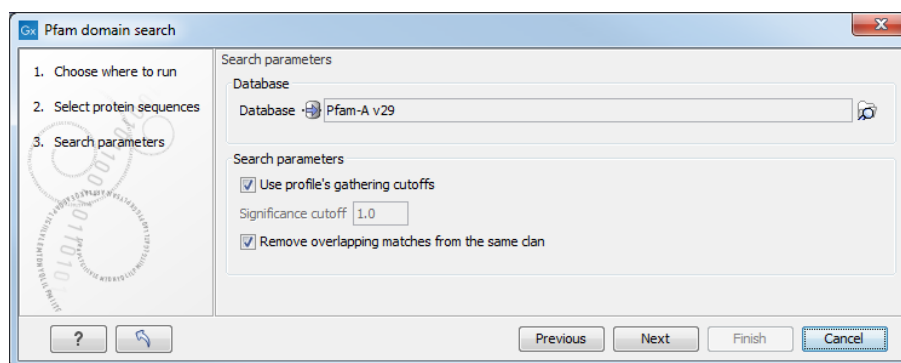


Figure 17.10: Setting parameters for Pfam Domain Search.

- **Use profile's gathering cutoffs.** Use cutoffs specifically assigned to each family by the curator instead of manually assigning the **Significance cutoff**.
- **Significance cutoff.** The E-value (expectation value) describes the number of hits one would expect to see by chance when searching a database of a particular size. Essentially, a hit with a low E-value is more significant compared to a hit with a high E-value. By lowering the significance threshold the domain search will become more specific and less sensitive, i.e. fewer hits will be reported but the reported hits will be more significant on average.
- **Remove overlapping matches from the same clan.** Perform post-processing of the results where overlaps between hits are resolved by keeping the hit with the smallest e-value.

Click **Next** to adjust the output of the tool. The Pfam search tool can produce two types of output. It can add annotations on the input sequences that show the domains found (see figure 17.11) and it can output a table with all the domains found.

Click **Finish** to start the tool.

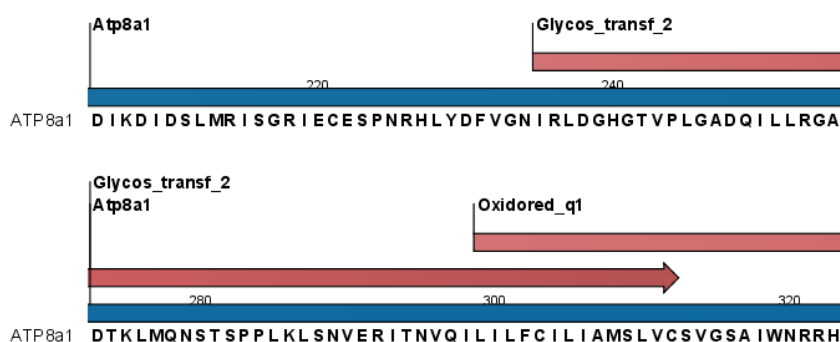


Figure 17.11: Annotations (in red) that were added by the Pfam search tool.

Domain annotations added by the Pfam search tool have the type **Region**. If the annotations are not visible they have to be enabled in the side panel. Detailed information for each domain annotation, such as the bit score which is the basis for the prediction of domains, is available through the annotation tool tip.

A more detailed description of the scores provided in the annotation tool tips can be found here: <http://pfam.sanger.ac.uk/help#tabview=tab5>.

17.5 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:

Toolbox | Protein Analysis (📁) | Predict secondary structure (🌀)

This opens the dialog displayed in figure 17.12:

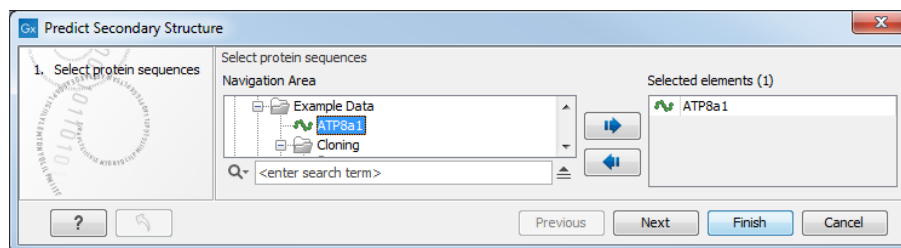


Figure 17.12: Choosing one or more protein sequences for secondary structure prediction.

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 **Finish** to start the tool.

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 17.13).

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 11.3.2.

Undesired alpha-helices or beta-sheets can be removed through the **Delete Annotation** (👉) right-click mouse menu. See section 11.3.4.

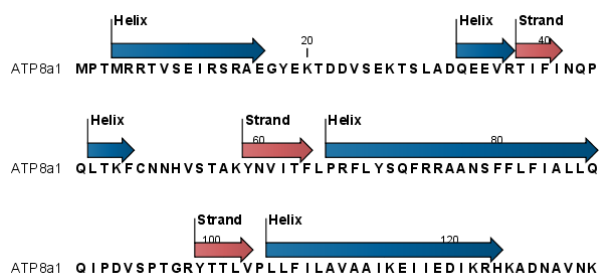


Figure 17.13: Alpha-helices and beta-strands shown as annotations on the sequence.

17.6 Protein report

CLC Main Workbench is able to produce protein reports, a collection of some of the protein analyses described elsewhere in this manual.

To create a protein report do the following:

Toolbox | Protein Analysis (📁) | Create Protein Report (📄)

This opens a dialog where you can choose which proteins to create a report for. If you had already selected a sequence in the Navigation Area before running the Toolbox action, this will be shown in the **Selected Elements**. However, you can use the arrows to change this. When the correct one is chosen, click **Next**.

In the next dialog, 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.** Will produce a section called Protein statistics, as described in section [15.6.1](#).
- **Protein charge plot.** Plot of charge as function of pH, see section [17.1](#).
- **Hydrophobicity plot.** See section [17.3](#).
- **Complexity plot.** See section [15.5](#).
- **Dot plot.** See section [15.4](#).
- **Secondary structure prediction.** See section [17.5](#).
- **Pfam domain search.** See section [17.4](#).
- **BLAST against one or more sequences.** See section [23.1.2](#).
- **BLAST against NCBI databases.** See section [23.1.1](#).

When you have selected the relevant analyses, click **Next**. In the following dialogs, adjust the parameters for the different analyses you selected. The parameters are explained in more details in the relevant chapters or sections (mentioned in the list above).

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.

For hydrophobicity plots:

- **Hydrophobicity scales.** Lets you choose between different scales.
- **Window size.** Width of window on sequence (it must be an odd number).

For complexity plots:

- **Window size.** Width of window on sequence (must be odd).

For dot plots:

- **Score model.** Different scoring matrices.
- **Window size.** Width of window on sequence.

For Pfam domain search:

- **Database and search type** lets you choose different databases and specify the search for full domains or fragments.
- **Significance cutoff** lets you set your E-value.

For BLAST search:

- **Program** lets you choose between different BLAST programs.
- **Target** lets you select the sequence(s) against which the BLAST should be targeted.
- **Database** lets you limit your search to a particular database.

Also set the BLAST parameters as explained in section [23.1.2](#).


For BLAST against NCBI databases:

- **Program** lets you choose between different BLAST programs.
- **Database** lets you limit your search to a particular database.
- **Genetic code** lets you choose a genetic code for the sequence or the database.

Also set the BLAST parameters as explained in section [23.1.1](#).

An example of Protein report can be seen in figure [17.14](#).

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**.

The content of the tables in the report can be copy/pasted out of the program and e.g. into Microsoft Excel. You can also **Export**  the report in Excel format.

1 Protein statistics

1.1 Sequence information

Sequence type	Protein
Length	1,149aa
Organism	Mus musculus
Name	ATP8a1
Description	Probable phospholipid-transporting ATPase IA (Chromaffin granule ATPase II) (ATPase class I type 8A member 1).
Modification Date	05-FEB-2008
Weight	129.765 kDa
Isoelectric point	7.22
Aliphatic index	99.347

1.2 Half-life

N-terminal aa	Half-life mammals	Half-life yeast	Half-life E.Coli
Methionine	30 hours	>20 hours	>10 hours

1.3 Extinction coefficient

Conditions	Extinction coefficient at 280nm	Absorption at 280nm 0.1% (=1 g/l)
Non-reduced cysteines	167,600	1.292
Reduced cysteines	166,280	1.281

Report Settings

Table of Contents

- 1 Protein statistics
 - 1.1 Sequence information
 - 1.2 Half-life
 - 1.3 Extinction coefficient
 - 1.4 Atomic composition
 - 1.5 Count of hydrophobic and hydrophilic residues
 - 1.6 Count of charged residues
 - 1.7 Amino acid distribution table
 - 1.8 Amino acid distribution histogram
 - 1.9 Annotation table
 - 1.10 Counts of di-peptides
 - 1.11 Frequency of di-peptides
- 2 Electrical charge as a function of pH
- 3 Plot of local Hydropathy
- 4 Plot of local sequence complexity
- 5 Dot plot of the sequence against itself
- 6 Secondary structure
- 7 Pfam result
- 8 BLAST against one or more sequences
- 9 BLAST against NCBI databases

Text format

Figure 17.14: A protein report. There is a Table of Contents in the Side Panel that makes it easy to browse the report.

17.7 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 17.7.1.

In order to make a reverse translation:

Toolbox | Protein Analysis (📁) | Reverse Translate (🔄)

This opens the dialog displayed in figure 17.15:

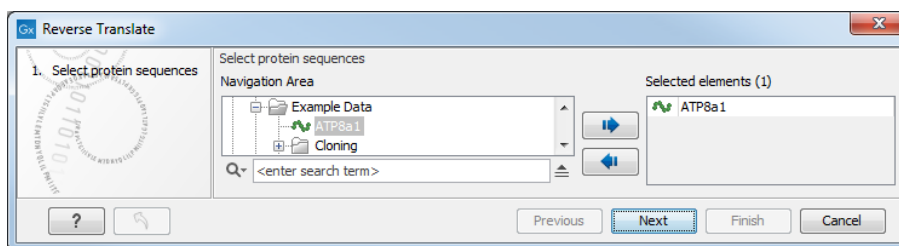


Figure 17.15: Choosing a protein sequence for reverse translation.

If a sequence was selected before choosing the Toolbox action, the sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements. You can translate several protein sequences at a time.

Adjust the parameters for the translation in the dialog shown in figure 17.16.

- **Use random codon.** This will randomly back-translate an amino acid to a codon assuming

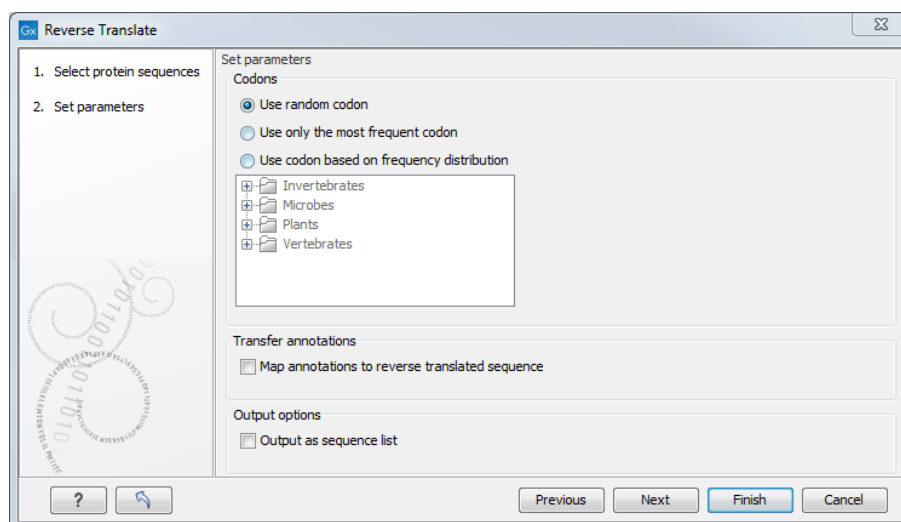


Figure 17.16: Choosing parameters for the reverse translation.

the genetic code to be 1, but without using the codon frequency 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 J.

Click **Finish** to start the tool. 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.

17.7.1 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 21st 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

17.8 Proteolytic cleavage detection

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 as well as in a table below the sequence view.

Detection of proteolytic cleavage sites is initiated by:

Toolbox | Protein Analysis (📁) | Proteolytic Cleavage (✂️)

This opens the dialog shown in figure 17.17. You can select one or several sequences.

In the second dialog, you can select proteolytic cleavage enzymes. Presently, the list contains the enzymes shown in figure 17.18. The full list of enzymes and their cleavage patterns can be seen in Appendix, section C.

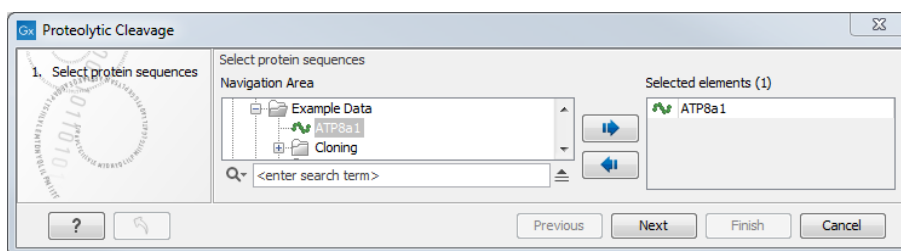


Figure 17.17: Choosing a protein sequence for proteolytic cleavage.

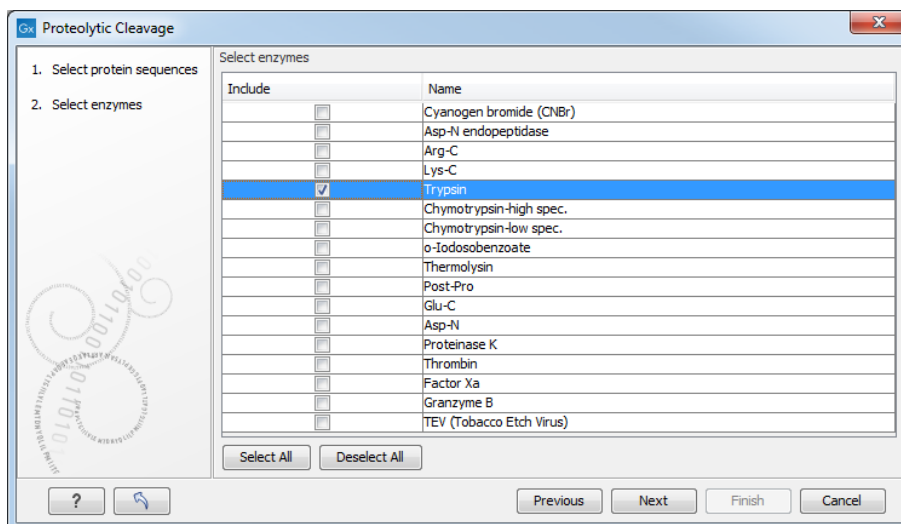


Figure 17.18: Setting parameters for proteolytic cleavage detection.

You can then set parameters for the detection. This limits the number of detected cleavages (figure 17.19).

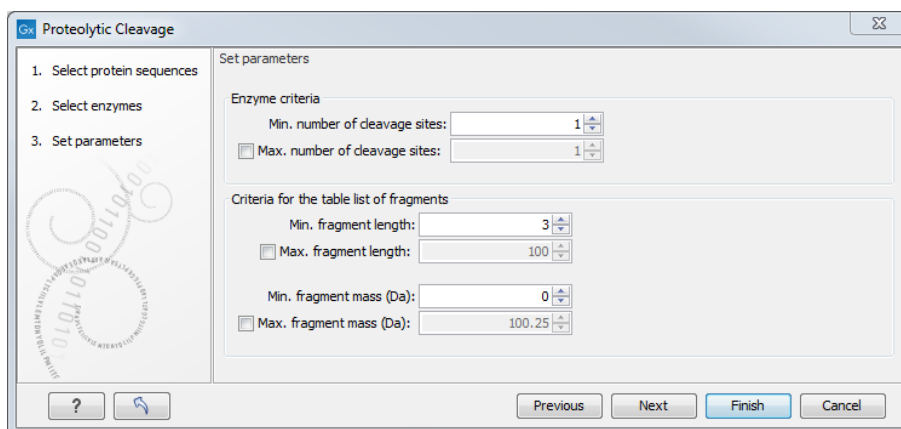


Figure 17.19: Setting parameters for proteolytic cleavage detection.

- **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.

For example, if you have one protein sequence but you only want to show which enzymes cut between two and four times. Then you should select "The enzymes has more cleavage sites than 2" and select "The enzyme has less cleavage sites than 4". In the next step you should simply select all enzymes. This will result in a view where only enzymes which cut 2,3 or 4 times are presented.

Click **Finish** to start the tool. The result of the detection is displayed in figure 17.20.

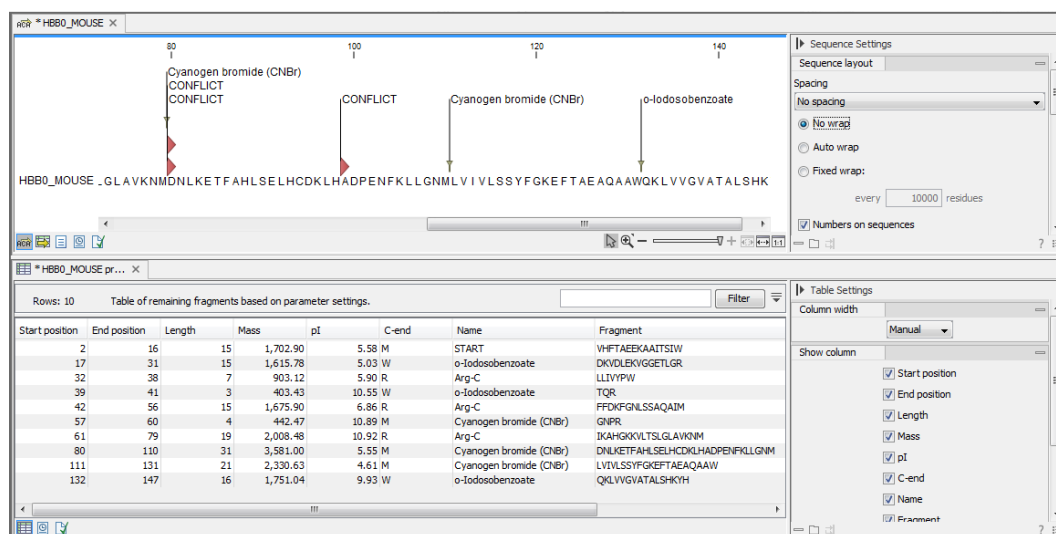


Figure 17.20: The result of the proteolytic cleavage detection.

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.

17.8.1 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 17.21.

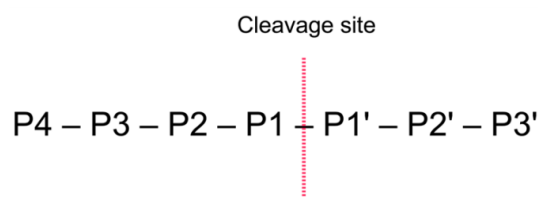


Figure 17.21: *Nomenclature of the peptide substrate. The substrate is cleaved between position P1-P1'.*

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 17.22).

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/>

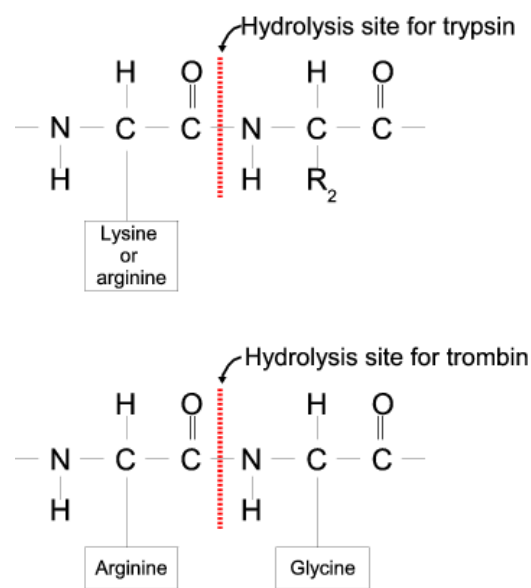


Figure 17.22: Hydrolysis of the peptide bond between two amino acids. Trypsin cleaves unspecifically at lysine or arginine residues whereas trombin cleaves at arginines if aspartate or glutamate is absent.

Chapter 18

Sequencing data analyses and Assembly

Contents

18.1 Importing and viewing trace data	349
18.1.1 Trace settings in the Side Panel	349
18.2 Trim sequences	350
18.2.1 Trimming using the Trim tool	351
18.2.2 Manual trimming	353
18.3 Assemble sequences	353
18.4 Assemble sequences to reference	355
18.5 Sort sequences by name	357
18.6 Add sequences to an existing contig	361
18.7 View and edit contigs and read mappings	362
18.7.1 View settings in the Side Panel	363
18.7.2 Editing a contig or read mapping	366
18.7.3 Sorting reads	366
18.7.4 Read conflicts	367
18.7.5 Using the mapping	367
18.7.6 Extract parts of a mapping	367
18.7.7 Variance table	369
18.8 Reassemble contig	371
18.9 Secondary peak calling	371

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 6.1).

This chapter first explains how to trim sequence reads. Next follows a description of how to assemble reads into contigs both with and without a reference sequence. In the final section, the options for viewing and editing contigs are explained.

18.1 Importing and viewing trace data

A number of different binary trace data formats can be imported into the program, including *Standard Chromatogram Format (.SCF)*, *ABI sequencer data files (.ABI and .AB1)*, *PHRED output files (.PHD)* and *PHRAP output files (.ACE)* (see section 6.1).

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.

You can see additional information about the quality of the traces by holding the mouse cursor on the imported sequence. This will display a tool tip as shown in figure 18.1.

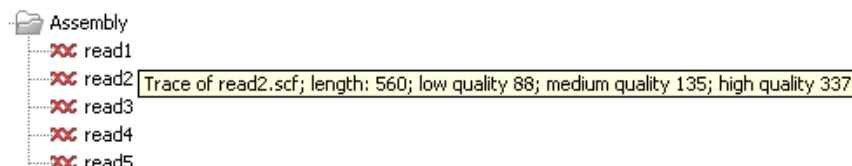


Figure 18.1: A tooltip displaying information about the quality of the chromatogram.

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 (ACT).

The traces can be scaled by dragging the trace vertically as shown in figure 18.2. The Workbench automatically adjust the height of the traces to be readable, but if the trace height varies a lot, this manual scaling is very useful.

The height of the area available for showing traces can be adjusted in the **Side Panel** as described in section 18.1.1.

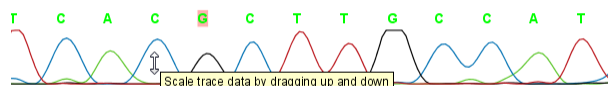


Figure 18.2: Grab the traces to scale.

18.1.1 Trace settings in the Side Panel

In the Nucleotide info preference group the display of trace data can be selected and unselected. When selected, the trace data information is shown as a plot beneath the sequence. The appearance of the plot can be adjusted using the following options (see figure 18.3):

- **Nucleotide trace.** For each of the four nucleotides the trace data can be selected and unselected.
- **Scale traces.** A slider which allows the user to scale the height of the trace area. Scaling the traces individually is described in section 18.1.

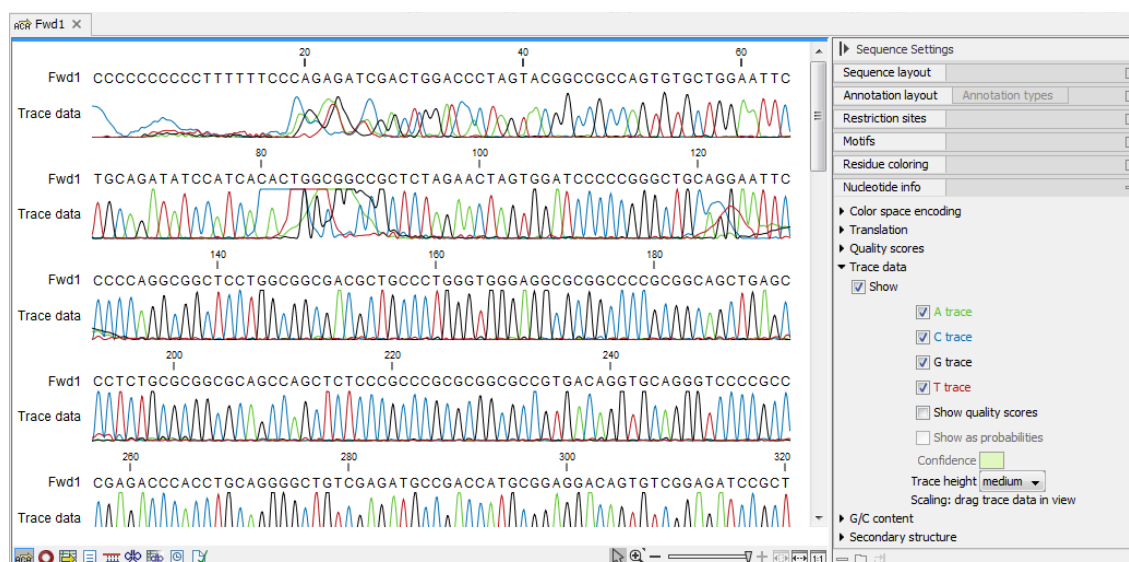


Figure 18.3: A sequence with trace data. The preferences for viewing the trace are shown in the Side Panel.

When working with stand-alone 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.

18.2 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 18.4). 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 18.3). 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 11.3.2).

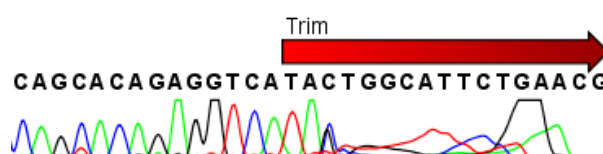


Figure 18.4: Trimming creates annotations on the regions that will be ignored in the assembly process.

When exporting sequences in fasta format, there is an option to remove the parts of the sequence covered by trim annotations.

18.2.1 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 | Sanger Sequencing Analysis (A) | Trim Sequences (S)

This opens a dialog where you can choose the sequences to trim, by using the arrows to move them between the Navigation Area and the 'Selected Elements' box.

You can then specify the trim parameters as displayed in figure 18.5.

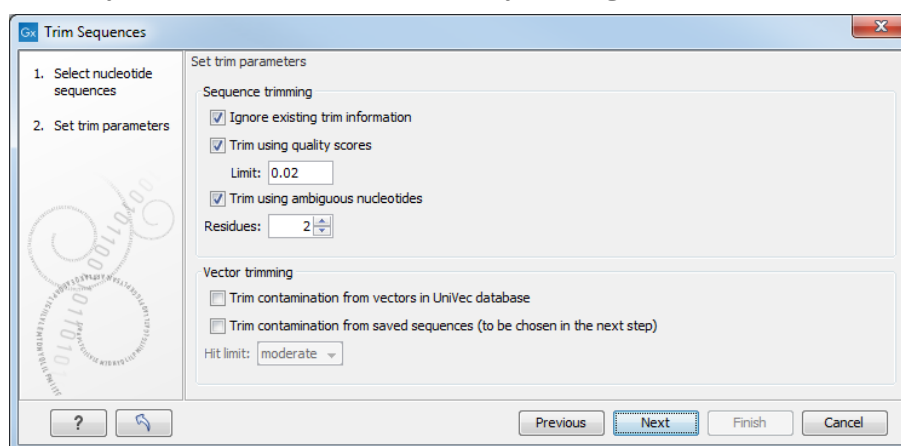


Figure 18.5: Setting parameters for trimming.

The following parameters can be adjusted in the dialog:

- **Ignore existing trim information.** If you have previously trimmed the sequences, you can check this to remove existing trimming annotation prior to analysis.
- **Trim using quality scores.** If the sequence files contain quality scores from a base caller algorithm this information can be used for trimming sequence ends. The program uses the modified-Mott trimming algorithm for this purpose (Richard Mott, personal communication): Quality scores in the Workbench are on a Phred scale, and formats using other scales will be converted during import. The Phred quality scores (Q), defined as: $Q = -10\log_{10}(P)$, 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 ending at the highest value of the running sum and starting at the last zero value before this highest score. Everything before and after this region will be trimmed. A read will be completely removed if the score never makes it above zero.

- **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. The "Trim ambiguous nucleotides" option trims all types of ambiguous nucleotides (see Appendix G).
 - **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 **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 **Finish** to start the tool. 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.

18.2.2 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.

18.3 Assemble sequences

This section describes how to assemble a number of sequence reads into a contig without the use of a reference sequence (a known sequence that can be used for comparison with the other sequences, see section 18.4).

Note! You can assemble a maximum of 10,000 sequences at a time.

To assemble more sequences, you need the *CLC Genomics Workbench* (see <http://www.qiagenbioinformatics.com/products/clc-genomics-workbench/>).

To perform the assembly:

Toolbox | Sanger Sequencing Analysis (AA) | Assemble Sequences (AA)

This will open a dialog where you can select sequences to assemble. If you already selected sequences in the Navigation Area, these will be shown in 'Selected Elements'. You can alter your choice of sequences to assemble, or add others, by using the arrows to move sequences between the Navigation Area and the 'Selected Elements' box. You can also add sequence lists.

When the sequences are selected, click **Next**. This will show the dialog in figure 18.6

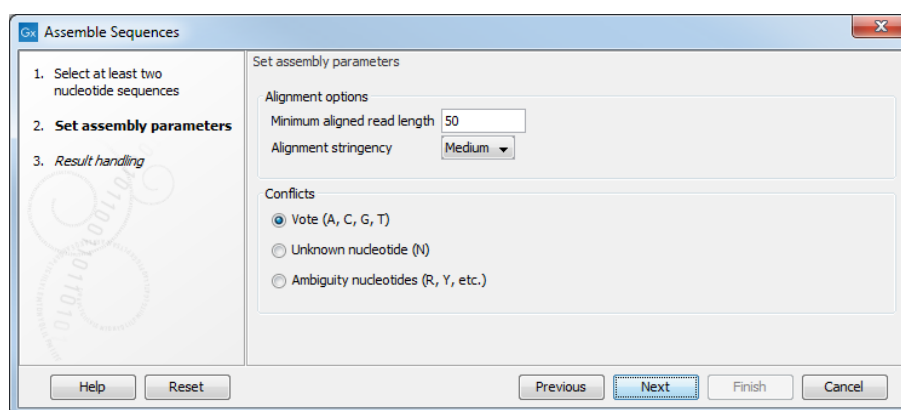



Figure 18.6: Setting assembly parameters.

This dialog gives you the following options for assembly:

- **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 (Low, Medium or High) of the scoring function used by the alignment step in the contig assembly algorithm. A higher stringency level will tend to produce contigs with fewer ambiguities but will also tend to omit more sequencing reads and to generate more and shorter contigs.
- **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 G.

Note, that conflicts will always be highlighted no matter which of the options you choose. Furthermore, each conflict will be marked as annotation on the contig sequence and will be present if the contig sequence is extracted for further analysis. As a result, the details of any experimental heterogeneity can be maintained and used when the result of single-sequence analyzes is interpreted. Read more about conflicts in section 18.7.4.

- **Create full contigs, including trace data.** This will create a contig where all the aligned reads are displayed below the contig sequence. (You can always extract the contig sequence without the reads later on.) For more information on how to use the contigs that are created, see section 18.7.
- **Show tabular view of contigs.** A contig can be shown both in a graphical as well as a tabular view. If you select this option, a tabular view of the contig will also be opened (Even if you do not select this option, you can show the tabular view of the contig later on by clicking **Table** () at the bottom of the view.) For more information about the tabular view of contigs, see section 18.7.7.
- **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.

When the assembly process has ended, a number of views will be shown, each containing a contig of two or more sequences that have been matched. If the number of contigs seem too high or low, try again with another **Alignment stringency** setting. Depending on your choices of output options above, the views will include trace files or only contig sequences. However, the calculation of the contig is carried out the same way, no matter how the contig is displayed.

See section 18.7 on how to use the resulting contigs.

18.4 Assemble sequences to reference

This section describes how to assemble a number of sequence reads into a contig using a reference sequence, a process called read mapping. A reference sequence can be particularly helpful when the objective is to characterize SNP variation in the data.

Note! You can assemble a maximum of 10,000 sequences at a time.

To assemble more sequences, you need the *CLC Genomics Workbench* (see <http://www.qiagenbioinformatics.com/products/clc-genomics-workbench/>).

To start the assembly:

Toolbox | Sanger Sequencing Analysis (A) | Assemble Sequences to Reference (A)

This opens a dialog where you can alter your choice of sequences to assemble. If you have already selected sequences in the Navigation Area, these will be shown in Selected Elements, however you can remove these or add others, by using the arrows to move sequences between the Navigation Area and Selected Elements boxes. You can also add sequence lists.

When the sequences are selected, click **Next**, and you will see the dialog shown in figure 18.7

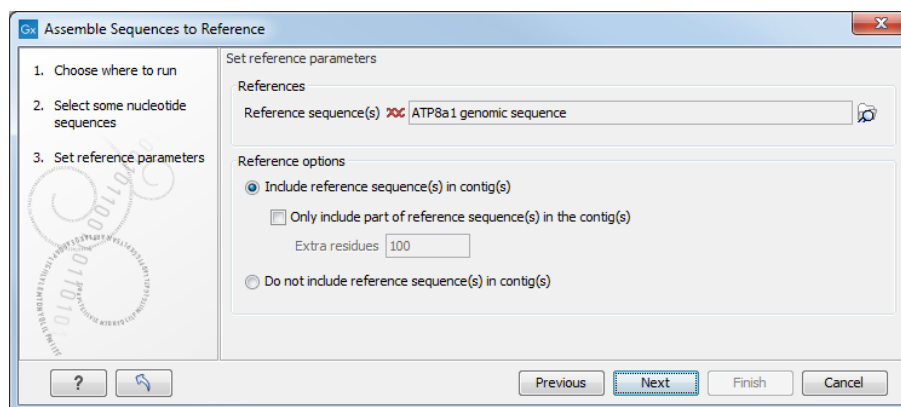


Figure 18.7: Parameters for how the reference should be handled when assembling sequences to a reference sequence.

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 18.8

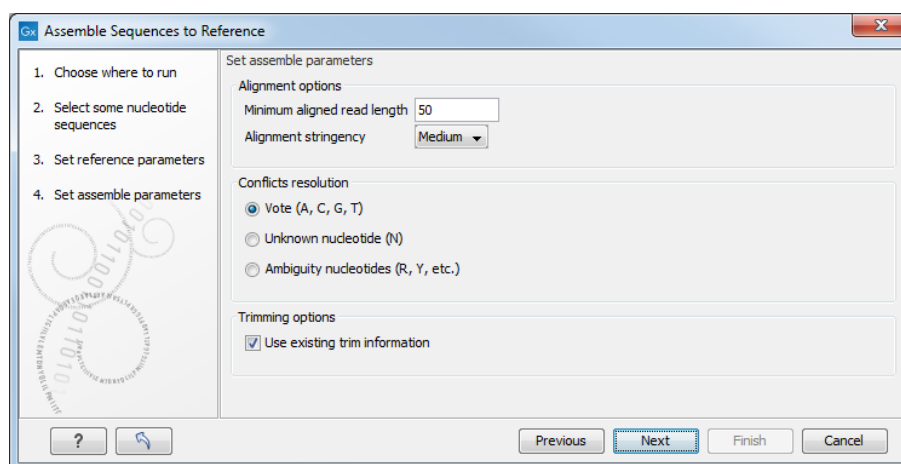


Figure 18.8: Options for how the input sequences should be aligned and how nucleotide conflicts should be handled.

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 (Low, Medium or High) 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.

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 G.
 - **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 18.2 for more information about trimming).

Click **Finish** to start the tool. This will start the assembly process. See section 18.7 on how to use the resulting contigs.

18.5 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  | **Sanger Sequencing Analysis**  | **Sort Sequences by Name** 

This opens a dialog where you can add the sequences you wish to sort, by using the arrows to move them between the Navigation Area and 'Selected Elements'. 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 18.9.

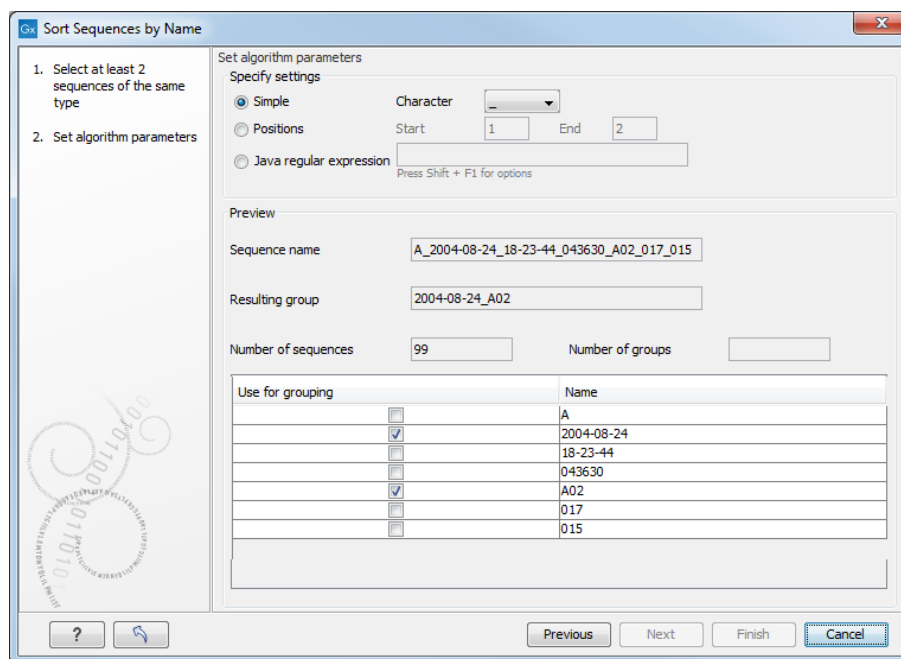


Figure 18.9: Splitting up the name at every underscore (`_`) and using the date and analysis position for grouping.

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 **Finish** to start the tool. 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 18.9.

Advanced splitting using regular expressions

You can see a more detail explanation of the regular expressions syntax in section 15.9.3.

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 18.9 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 18.10.

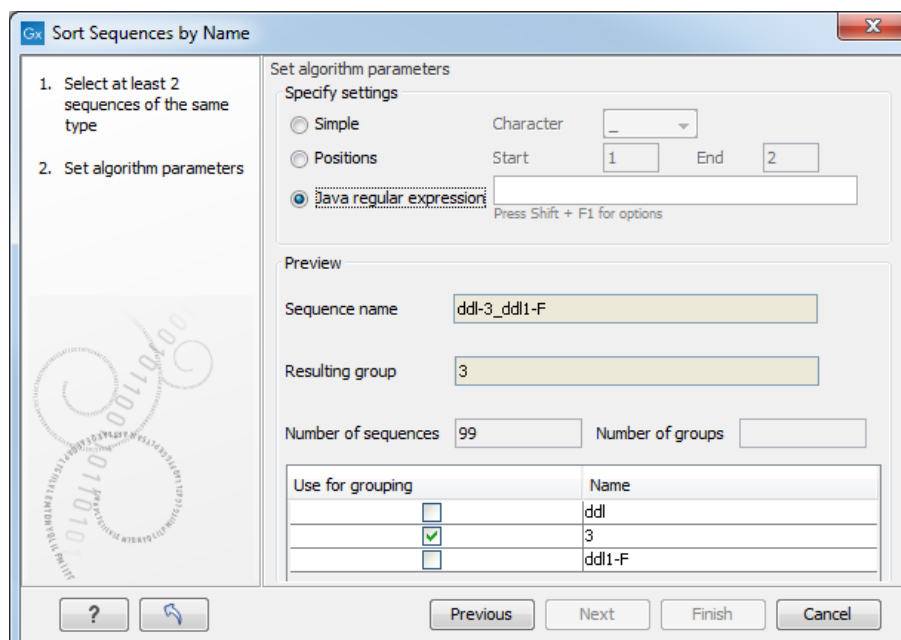


Figure 18.10: Dividing the sequence into three groups based on the number in the middle of the name.

The round brackets () denote the part of the name that will be listed in the groups table at the bottom of the dialog. In this example we actually did not need the first and last set of brackets, so the expression could also have been `. * - (. *) _ . *` in which case only one group would be listed in the table at the bottom of the dialog.

18.6 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:

Toolbox in the Menu Bar | Sanger Sequencing Analysis (A) | Add Sequences to Contig (A)

or **right-click in the empty white area of the contig | Add Sequences to Contig (A)**

This opens a dialog where you can select one contig and a number of sequences to assemble. If you have already selected sequences in the Navigation Area, these will be shown in the 'Selected Elements' box. However, you can remove these, or add others, by using the arrows to move sequences between the Navigation Area and Selected Elements boxes. You can also add sequence lists.

Often, the results of the assembly will be better if the sequences are trimmed first (see section 18.2.1).

When the elements are selected, click **Next**, and you will see the dialog shown in figure 18.11

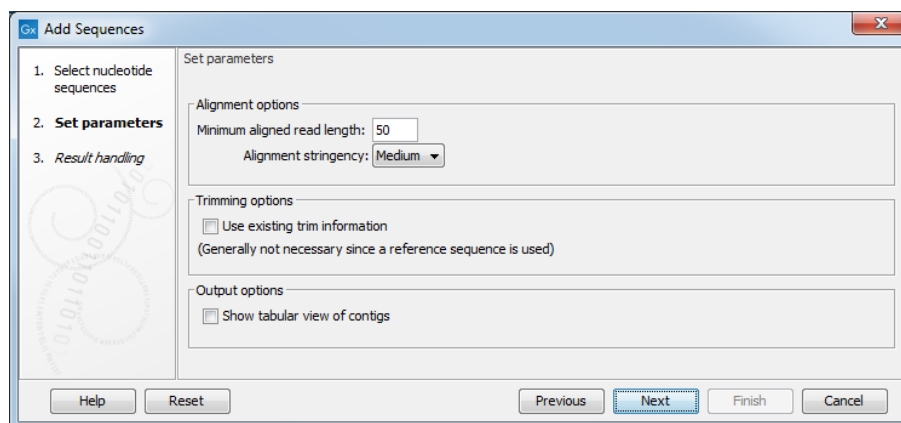


Figure 18.11: Setting assembly parameters when assembling to an existing contig.

The options in this dialog are similar to the options that are available when assembling to a reference sequence (see section 18.4).

Click **Finish** to start the tool. This will start the assembly process. See section 18.7 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.

18.7 View and edit contigs and read mappings

The results of the assembly or mapping (assembly to a reference) are respectively contigs or a read mapping. In both cases the sequence reads have been aligned (see figure 18.12). If multiple reference sequences were used, this information will be in a table where the actual visual mapping can be opened by double-clicking.

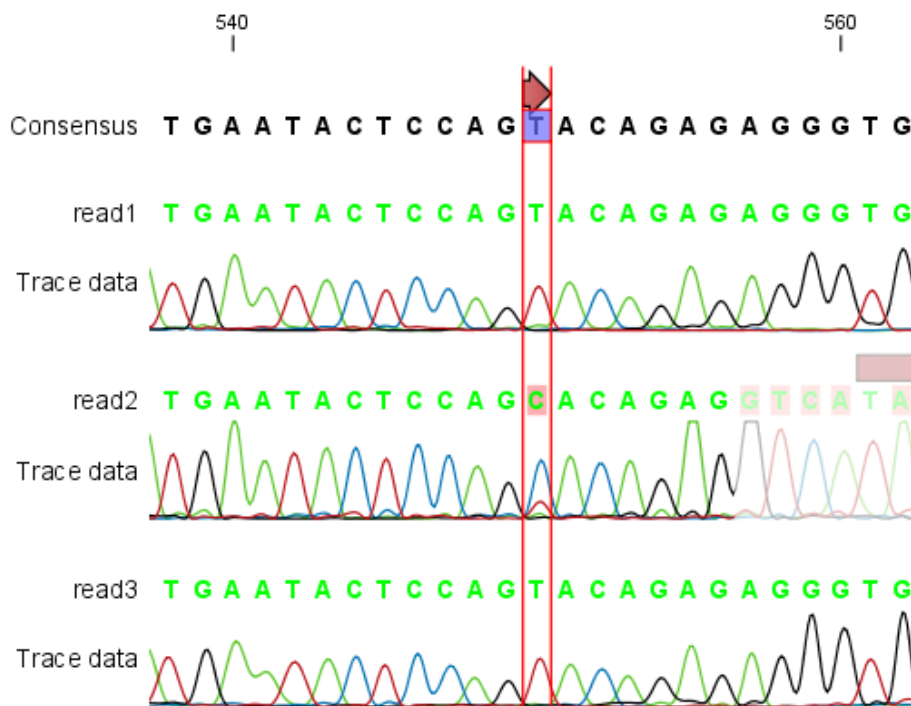


Figure 18.12: *The view of a contig. Note that you can zoom to a very detailed level.*

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 or mapping. This may be due to trimming before or during the assembly or to misalignment to the other reads.

You can easily adjust the trimmed area to include more of the read in the contig or mapping: simply drag the edge of the faded area as shown in figure 18.13.



Figure 18.13: *Dragging the edge of the faded area.*

Note! The handles for dragging are only available at the zoom level where residues can be seen. This means that you need to have zoomed in to 100% or more and chosen **Compactness** levels "Not compact", "Low" or "Packed".

Residues are colored green unless they were reversed to map. In which case they will be red. The colors can be changed in the **Side Panel** as described in section 18.7.1

If you find out that the reversed reads should have been the forward reads and vice versa, you can reverse complement the whole contig or mapping. Right-click in the empty white area of the contig or mapping and choose to **Reverse Complement Sequence**.

18.7.1 View settings in the Side Panel

The View Settings panel for assemblies and read mappings with fewer than 2000 reads resembles that of alignments (see section 13.2) but has some extra preferences described below (figure 18.14).

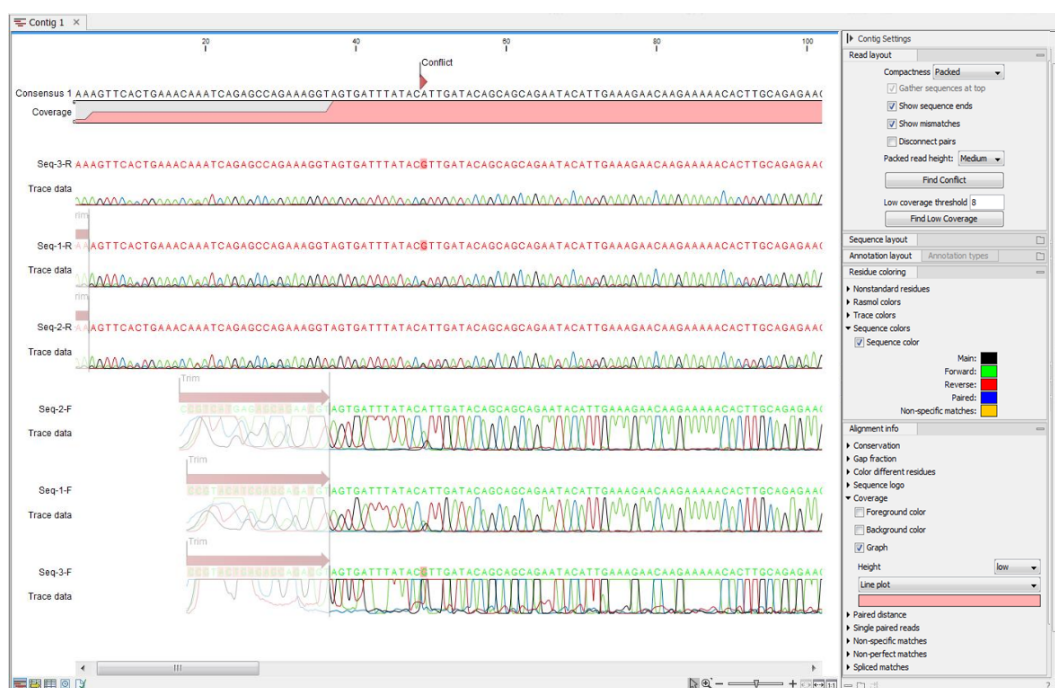


Figure 18.14: An example of contig, result of an assembly with less than 2000 reads

- **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" palette 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 in 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 be selected. For further details on these, please see section 18.1.1 and section 11.1.
 - * **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.
- **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 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 vertical lines. 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), meaning that 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.** Residues that are different from the reference are colored (as default), providing an overview of the conflicts. Clicking this button selects the next position

where there is an conflict between the sequence reads. 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 mapping. The level of coverage is relative to the overall number of sequence reads.
 - * **Foreground color.** Colors the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage.
 - * **Background color.** Colors the background of the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage
 - * **Graph.** The coverage is displayed as a graph (Learn how to export the data behind the graph in section 6.4).
 - **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.**
 - **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 specific elements of a contig or a mapping, which can be adjusted in the View settings. Those covered here were the key ones relevant for standard review of the results.

18.7.2 Editing a contig or read mapping


When editing contigs and read mappings, you are typically interested in confirming or changing single bases, and this can be done simply by:

selecting the base | typing the right base

Some users prefer to use lower-case letters in order to be able to see which bases were altered when they use the results later on. In *CLC Main Workbench* all changes are recorded in the history log (see section 2.1.2), 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 or mapping view, you can use **Zoom in**  to zoom to a greater level of detail than in other views (see figure 18.12). 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 or mappings with more than 1,000 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.

18.7.3 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.

18.7.4 Read conflicts

After assembly or mapping, 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 has a different residue compared to the reference*.

A conflict can be in two states:

- **Conflict.** Both the annotation and the corresponding row in the Table  are colored **red**.
- **Resolved.** Both the annotation and the corresponding row in the Table  are colored **green**.


The conflict can be resolved by correcting the deviating residues in the reads as described above.

A fast way of making all the reads reflect the consensus sequence is to select the position in the consensus, right-click the selection, and choose **Transfer Selection to All Reads**.


The opposite is also possible: make a selection on one of the reads, right click, and **Transfer Selection to Contig Sequence**.

18.7.5 Using the mapping

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 mapping and will be saved together with the mapping. It also means that if you add annotations to the sequence, they will be shown in the mapping view as well. This can be very convenient e.g. for Primer design .

If you wish to BLAST the consensus sequence, simply select the whole contig for your BLAST search. It will automatically extract the consensus sequence and perform the BLAST search.

In order to preserve the history of the changes you have made to the contig, the contig itself should be saved from the contig view, using either the save button  or by dragging it to the **Navigation Area**.

18.7.6 Extract parts of a mapping

Sometimes it is useful to extract part of a mapping for in-depth analysis. This could be the case if you have performed an 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 mapping data by running the **Extract from Selection** tool on a selected region in your mapping. The result of running this tool is a new mapping 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 2.2.3 for a detailed description of the different modes) and select your region of interest in your mapping, then right-click. You are now presented with the dialog shown in figure 18.15.

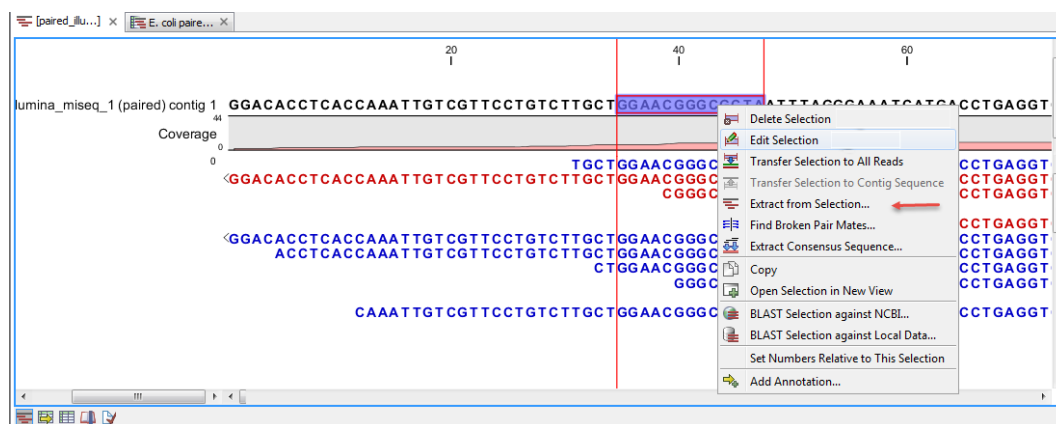


Figure 18.15: Extracting parts of a mapping.

When you choose the **Extract from Selection** option you are presented by the dialog shown in figure 18.16.

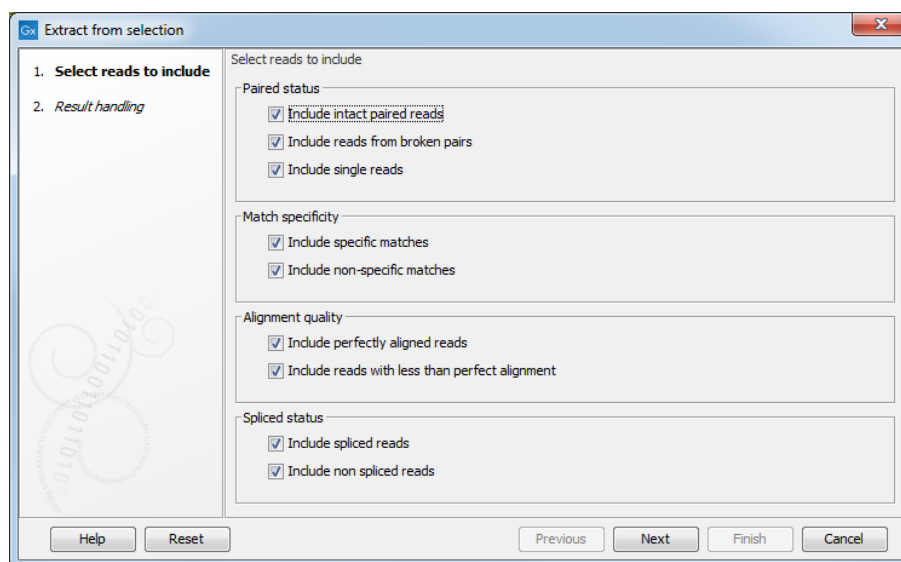


Figure 18.16: Selecting the reads to include.

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 distance specified, they will fall into this category. Per default, these reads are colored in blue.

Include paired reads from broken pairs When a pair is broken, either because only one read in the pair matches, or because the distance or relative orientation is wrong, the reads are placed and colored as single reads, but you can still extract them by checking this box.

Include single reads This will include reads that are marked as single reads (as opposed to paired reads). Note that paired reads that have been broken during assembly are not included in this category. Single reads that come from trimming paired sequence lists are included in this category.

Match specificity **Include specific matches** Reads that only are mapped to one position.

Include non-specific matches Reads that have multiple equally good alignments to the reference. These reads are colored yellow per default.

Alignment quality **Include perfectly aligned reads** Reads where *the full read* is perfectly aligned to the reference sequence (or consensus sequence for de novo assemblies). Note that at the end of the contig, reads may extend beyond the contig (this is not visible unless you make a selection on the read and observe the position numbering in the status bar). Such reads are not considered perfectly aligned reads because they don't align in their entire length.

Include reads with less than perfect alignment Reads with mismatches, insertions or deletions, or with unaligned nucleotides at the ends (the faded part of a read).

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 15.2)

You will now have all paired reads from the original mapping in a list.

18.7.7 Variance table

In addition to the standard graphical display of a contig or mapping as described above, you can also see a tabular overview of the conflicts between the reads by clicking the **Table**  icon at the bottom of the view.

This will display a new view of the conflicts as shown in figure 18.17.

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.



Figure 18.17: The graphical view is displayed at the top, and underneath the conflicts are shown in a table. At the conflict at position 313, the user has entered a comment in the table (to see it, make sure the Notes column is wide enough to display all text lines). This comment is now also added to the tooltip of the conflict annotation in the graphical view above.

- **Other residues.** Lists the residues of the reads. Inside the brackets, you can see the number of reads having this residue at this position. In the example in figure 18.17, 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 G.)
- **Status.** The status can either be conflict or resolved:
 - **Conflict.** Initially, all the rows in the table have this status. This means that there is one or more differences between the sequences at this position.
 - **Resolved.** If you edit the sequences, e.g. if there was an error in one of the sequences, and they now all have the same residue at this position, the status is set to *Resolved*.
- **Note.** Can be used for your own comments on this conflict. Right-click in this cell of the table to add or edit the comments. The comments in the table are associated with the conflict annotation in the graphical view. Therefore, the comments you enter in the table will also be attached to the annotation on the consensus sequence (the comments can be displayed by placing the mouse cursor on the annotation for one second - see figure 18.17). The comments are saved when you **Save** (⌘S).

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 or the mapping, as are using the **Find Conflict** button or using the **Space bar**. You can use the up and down arrow keys to navigate the rows of the table.

18.8 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 | Sanger Sequencing Analysis (🔍) | Reassemble Contig (🔄) | select the contig from Navigation Area, move to 'Selected Elements' and click Next

or **right-click in the empty white area of the contig | Reassemble contig (🔄)**

This opens a dialog as shown in figure 18.18

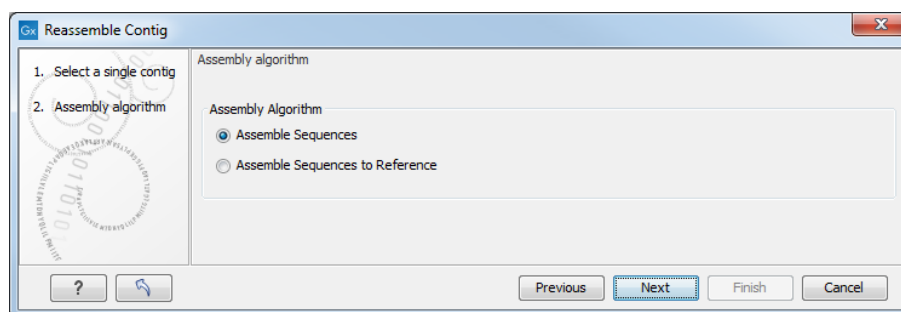


Figure 18.18: Re-assembling a contig.

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 18.3. The consensus sequence of the contig will be ignored.
- **Reference assembly.** This will use the consensus sequence of the contig as reference. When you click **Next**, you will follow the same steps as described in section 18.4.

When you click **Finish**, a new contig is created, so you do not lose the information in the old contig.

18.9 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). **Note!** The secondary peak caller does not call and annotate secondary peaks that have already been called by the Sanger sequencing machine and denoted with an ambiguity code.

Regions that are trimmed (i.e. covered by trim annotations) are ignored in the analysis (section 18.2).

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:

Toolbox | Sanger Sequencing Analysis (A) | Call Secondary Peaks (A)

This opens a dialog where you can add the sequences to be analyzed. If you had already selected sequence in the Navigation Area, these will be shown in the 'Selected Elements' box. However you can remove these, or add others, by using the arrows to move sequences between the Navigation Area and Selected Elements boxes.

When the sequences are selected, click **Next**.

This opens the dialog displayed in figure 18.19.

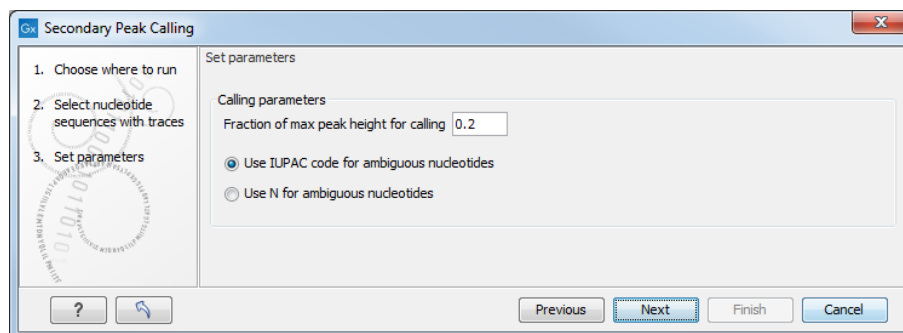


Figure 18.19: Setting parameters secondary peak calling.

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 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 G).

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 **Finish** to start the tool. This will start the secondary peak calling. A detailed history entry will be added to the history specifying all the changes made to the sequence.

Chapter 19

Primers and probes

Contents

19.1 Primer design - an introduction	374
19.1.1 General concept	374
19.1.2 Scoring primers	376
19.2 Setting parameters for primers and probes	376
19.2.1 Primer Parameters	376
19.3 Graphical display of primer information	378
19.3.1 Compact information mode	379
19.3.2 Detailed information mode	379
19.4 Output from primer design	380
19.5 Standard PCR	382
19.5.1 When a single primer region is defined	382
19.5.2 When both forward and reverse regions are defined	383
19.5.3 Standard PCR output table	384
19.6 Nested PCR	385
19.7 TaqMan	387
19.8 Sequencing primers	388
19.9 Alignment-based primer and probe design	389
19.9.1 Specific options for alignment-based primer and probe design	389
19.9.2 Alignment based design of PCR primers	391
19.9.3 Alignment-based TaqMan probe design	392
19.10 Analyze primer properties	394
19.11 Find binding sites and create fragments	395
19.11.1 Binding parameters	396
19.11.2 Results - binding sites and fragments	397
19.12 Order primers	399

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.

19.1 Primer design - an introduction

Primer design can be accessed in two ways:

Toolbox | Primers and Probes () | Design Primers () | OK

or **right-click sequence in Navigation Area | Show | Primer Designer ()**

In the primer view (see figure 19.1), the basic options for viewing the template sequence are the same as for the standard sequence view (see section 11.1 for an explanation of these options). This means that annotations such as known SNPs or exons can be displayed on the template sequence to guide the choice of primer regions. In addition, traces in sequencing reads can be shown along with the structure to guide the re-sequencing of poorly resolved regions.

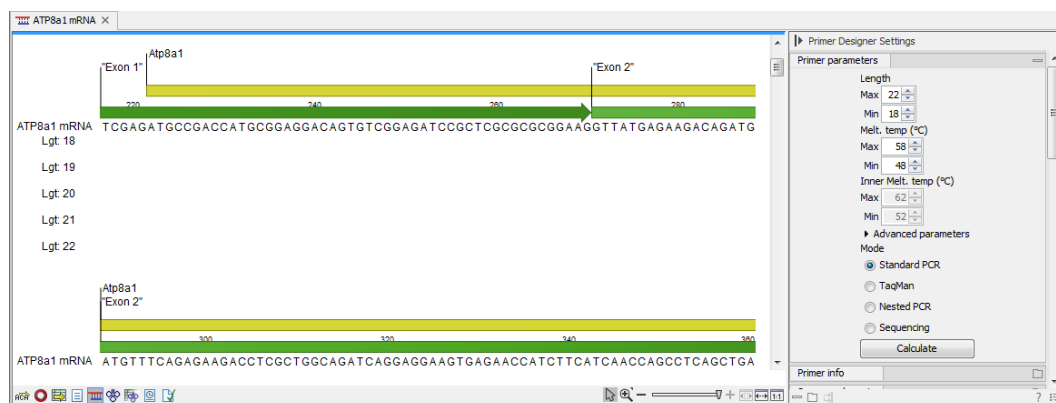


Figure 19.1: The initial view of the sequence used for primer design.

19.1.1 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 possible 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 19.2).

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 shown 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 fulfills 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

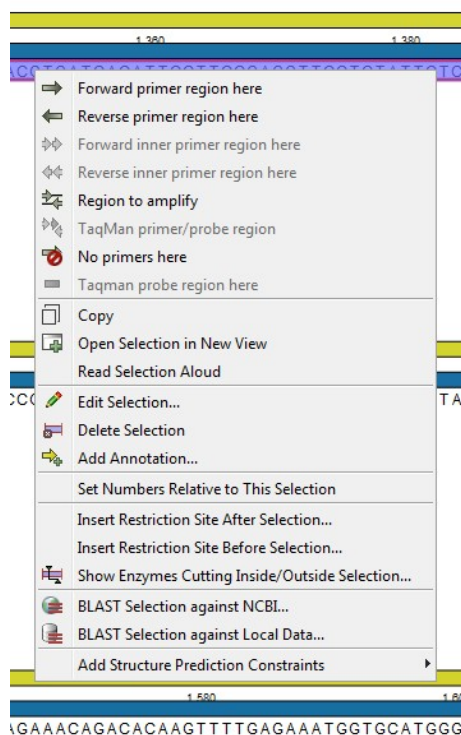


Figure 19.2: Right-click menu allowing you to specify regions for the primer design

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.

19.1.2 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.

19.2 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 19.3).

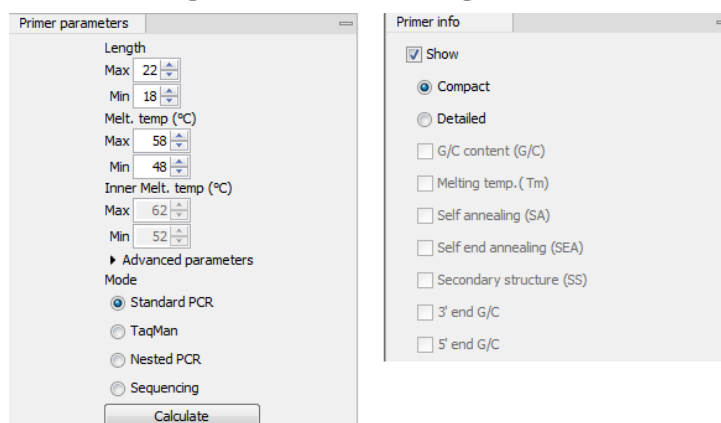


Figure 19.3: The two groups of primer parameters (in the program, the Primer information group is listed below the other group).

19.2.1 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). Note that in the case of a mix of primers, the concentration here refers to the individual primer and not the combined primers concentration.
 - * **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 GC 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

19.3 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 19.3).

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.

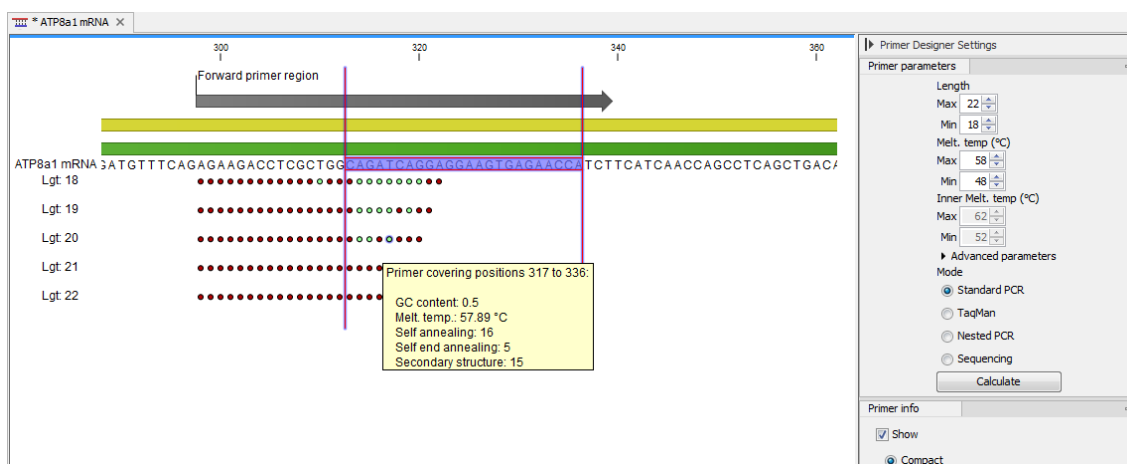


Figure 19.4: Compact information mode.

19.3.1 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 19.4).

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 fulfills 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, red circles will appear at the starting points of the primers which fail to meet this requirement.

19.3.2 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 19.5).

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:

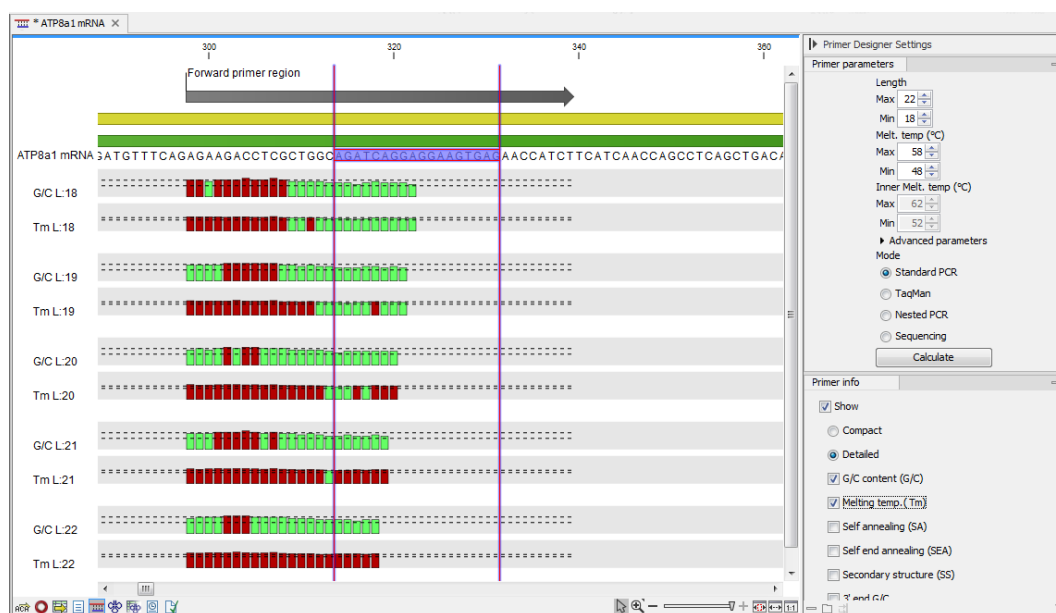


Figure 19.5: Detailed information mode.

- 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.

19.4 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 19.6).

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 screenshot displays two windows from a primer design software. The top window, titled '* ATP8a1 mRNA', shows a sequence editor with a yellow bar indicating a 'Forward primer region' on the ATP8a1 mRNA sequence. The sequence is: 3' ATGTTTCAGAGAAAGACCTCGCTGGCAGATCAGGAGGAAGTGAGAACCATCTTCATCAACCAGCCTCAGCTGAC. Below the sequence, primer lengths (Lgt) from 18 to 22 are shown with corresponding dot patterns. The bottom window, titled '* ATP8a1 mRNA p...', shows a table of proposed primers. The table has columns for Score, Sequence, Region, GC content, Melt. temp., and Secondary structure. Three primers are listed, each with a secondary structure diagram. The right side of the interface shows 'Primer Designer Settings' and 'Primer Table Settings' panels.

Score	Sequence	Region	GC content	Melt. temp.	Secondary struc...	Secondary struc...
42.82	AGATCAGGAGGAAGTGAGAA	Fwd (314, 333)	0.45	54.47	11.00	
42.82	GATCAGGAGGAAGTGAGAA	Fwd (315, 333)	0.47	52.67	11.00	
42.82	ATCAGGAGGAAGTGAGAA	Fwd (316, 333)	0.44	51.44	11.00	

Figure 19.6: Proposed primers.

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.

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. Furthermore, the primers can be edited using the standard sequence view to introduce e.g. mutations and restriction sites.

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.

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**.

19.5 Standard PCR

This mode is used to design primers for a PCR amplification of a single DNA fragment.

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 19.3) and setting the desired parameter values in the Primer Parameters preference group, the **Calculate** button will activate the primer design algorithm.

19.5.1 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 19.7).

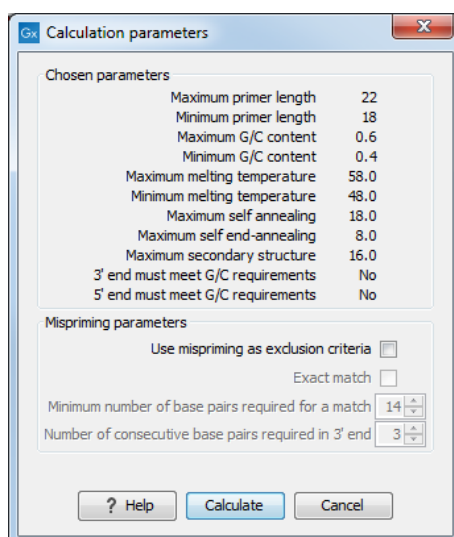


Figure 19.7: Calculation dialog for PCR primers when only a single primer region has been defined.

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.

Mispriming: The lower part contains a menu where the user can choose to include mispriming as an exclusion criteria in the design process. If this option is selected the algorithm will search for competing binding sites of the primer within the rest of the sequence, to see if the primer would match to multiple locations. If a competing site is found (according to the parameters set), the primer will be excluded.

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.

19.5.2 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 19.8).

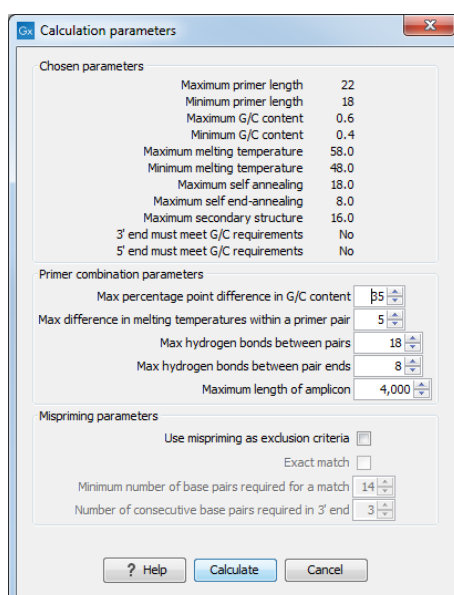


Figure 19.8: Calculation dialog for PCR primers when two primer regions have been defined.

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 section 19.5.1). 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.

19.5.3 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

19.6 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 19.3) 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 19.9).

The top and bottom parts of this dialog are identical to the *Standard PCR* dialog for designing primer pairs described above.

The central part of the dialog contains parameters pertaining to primer pairs and the comparison between the outer and the inner pair. Here five options can be set:

- Maximum percentage point difference in G/C content (described above under Standard PCR) - this criteria is applied to both primer pairs independently.
- 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

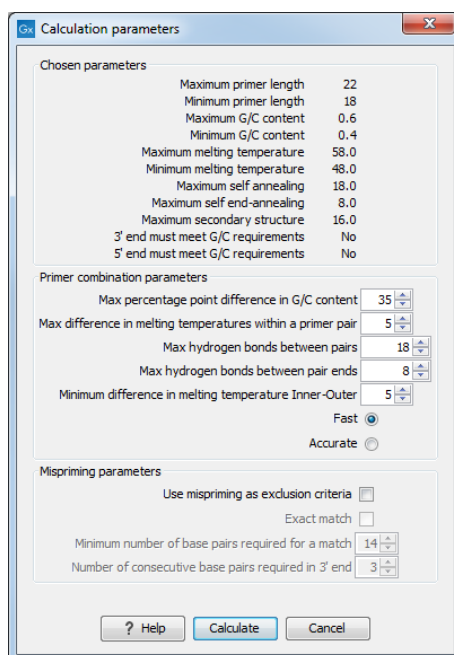


Figure 19.9: Calculation dialog for nested primers.

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.

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.

19.7 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 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 19.3) 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 19.10) which is similar to the *Nested PCR* dialog described above (see section 19.6).

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.

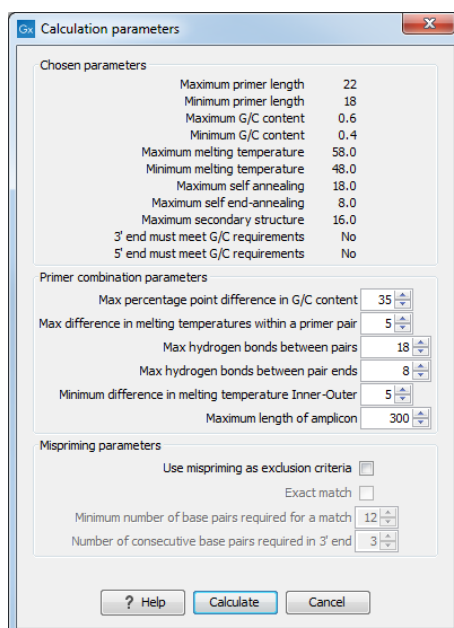


Figure 19.10: Calculation dialog for taqman primers.

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.

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.

19.8 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 19.3) 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 19.11).

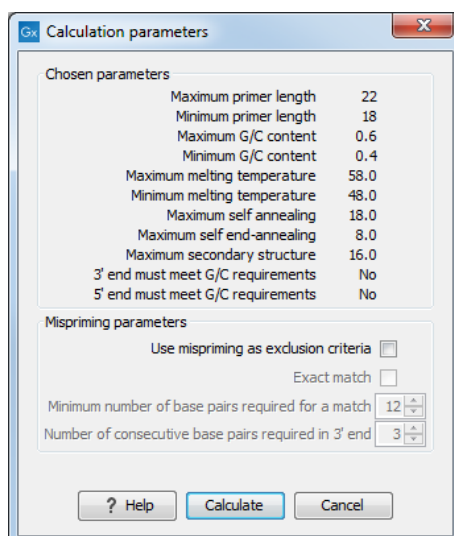


Figure 19.11: Calculation dialog for sequencing primers.

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 section 19.5 for a description).

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.

19.9 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 with:

Toolbox | Primers and Probes (🔍) | Design Primers (🔍)

Or if the alignment is already open, click Primer Designer (🔍) in the lower left part of the view.

In the alignment primer view (see figure 19.12), the basic options for viewing the template alignment are the same as for the standard view of alignments (see section 13 for an explanation of these options). This means that annotations such as known SNPs or exons can be displayed on the template sequence to guide the choice of primer regions.

19.9.1 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.

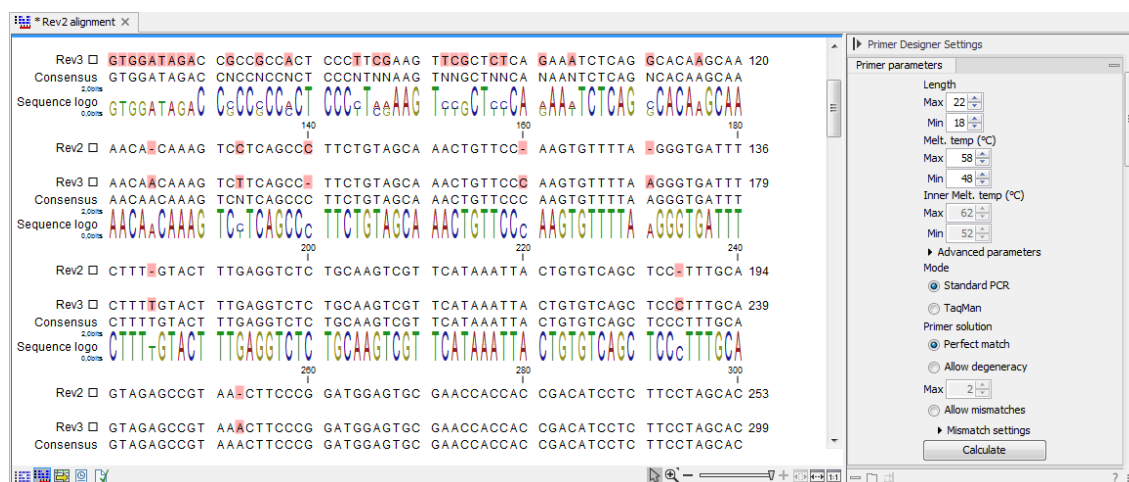


Figure 19.12: The initial view of an alignment used for primer design.

The **Primer Parameters** group in the **Side Panel** has the same options as the ones defined for primers design based on single sequences, but differs by the following submenus (see figure 19.12):

- In the **Mode** submenu, specify either:
 - **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.
- In the **Primer solution** submenu, 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 workflow when designing alignment based primers and probes is as follows (see figure 19.12):

- 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.

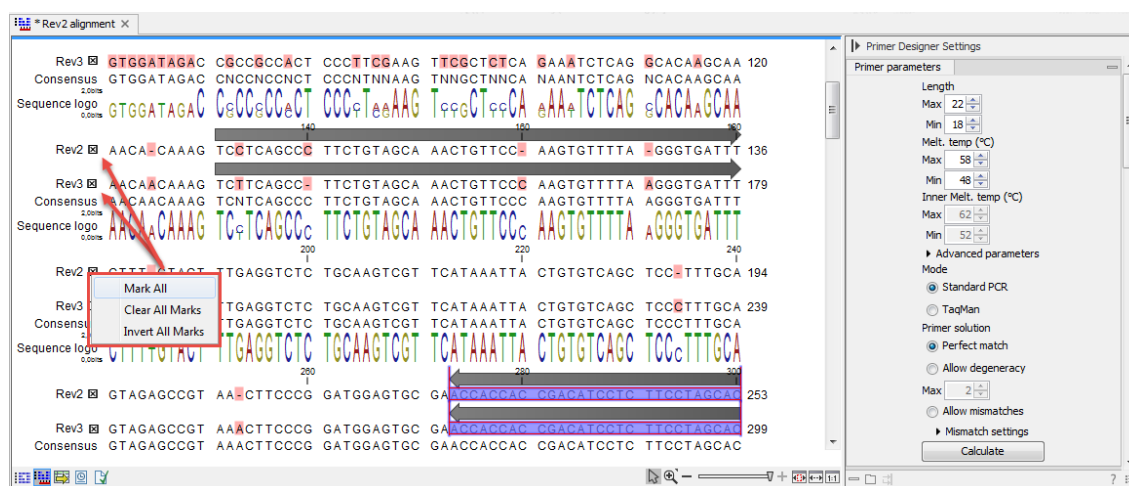


Figure 19.13: The initial view of an alignment used for primer design.

19.9.2 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 heterogeneities occur in the included template sequences. The allowed fold of degeneracy is user defined and corresponds to the number of possible primer combinations formed by a degenerate primer. Thus, if a primer covers two 4-fold degenerate site and one 2-fold degenerate site the total fold of degeneracy is $4 * 4 * 2 = 32$ and the primer will, when supplied from the manufacturer, consist of a mixture of 32 different oligonucleotides. When scoring the available primers, degenerate primers are given a score which decreases with the fold of degeneracy.
- **Allow mismatches.** Designs primers which are allowed a specified number of mismatches to the included template sequences. The melting temperature algorithm employed includes the latest thermodynamic parameters for calculating T_m when single-base mismatches occur.

When in Standard PCR mode, clicking the **Calculate** button will prompt the dialog shown in figure 19.14.

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.

19.9.3 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.

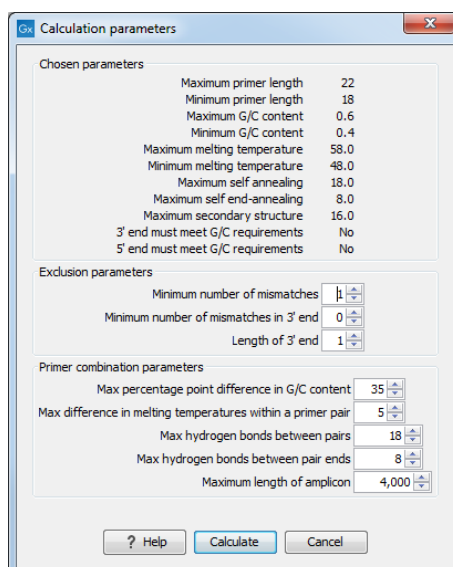


Figure 19.14: Calculation dialog shown when designing alignment based PCR primers.

Pushing the **Calculate** button will cause the dialog shown in figure 19.15 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.

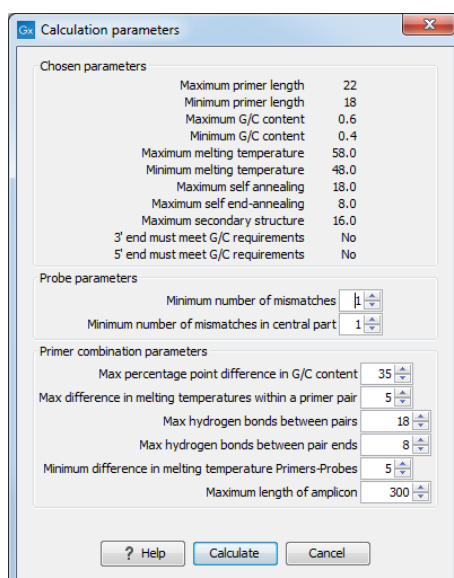


Figure 19.15: Calculation dialog shown when designing alignment based TaqMan probes.

19.10 Analyze primer properties

CLC Main Workbench can calculate and display the properties of predefined primers and probes:

Toolbox | Primers and Probes (📁) | Analyze Primer Properties (🔍)

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. (Primers are represented as DNA sequences in the Navigation Area).

Clicking **Next** generates the dialog seen in figure 19.16:

In the *Concentrations* panel a number of parameters can be specified concerning the reaction mixture and which influence melting temperatures

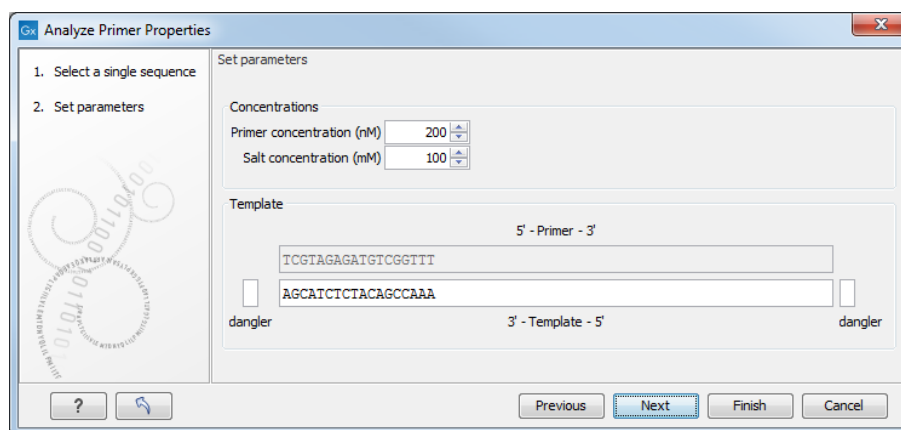


Figure 19.16: The parameters for analyzing primer properties.

- **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 **Finish** to start the tool. The result is shown in figure 19.17:

Sequence	Melt. temp.	Self annealing	Self annealing alignment	Self end-ann...	GC content	Secondary st...	Secondary s...
TCGTAGAGATGTCGGTTT	54.40	14.00	<pre> TCGTAGAGATGTCGGTTT TTGGCTGTAGAGATGCT </pre>	2.00	44.44	10.00	

Figure 19.17: Properties of a primer.

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 19.5.3.

19.11 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 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. Note that this tool is not meant to analyze rapidly high-throughput data. The maximum amount of sequences the tool will handle in a reasonable amount of time depends on your computer processing capabilities.

To search for primer binding sites:

Toolbox | Primers and Probes (📁) | Find Binding Sites and Create Fragments (🔍)

If a sequence was already selected in the Navigation Area, 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.

19.11.1 Binding parameters

This opens the dialog displayed in figure 19.18:

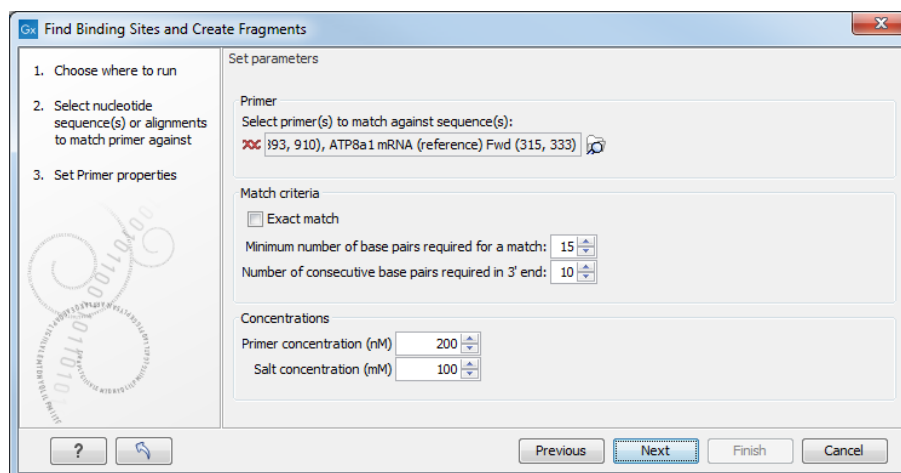


Figure 19.18: Search parameters for finding primer binding sites.

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. 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)

19.11.2 Results - binding sites and fragments

Specify the output options as shown in figure 19.19:

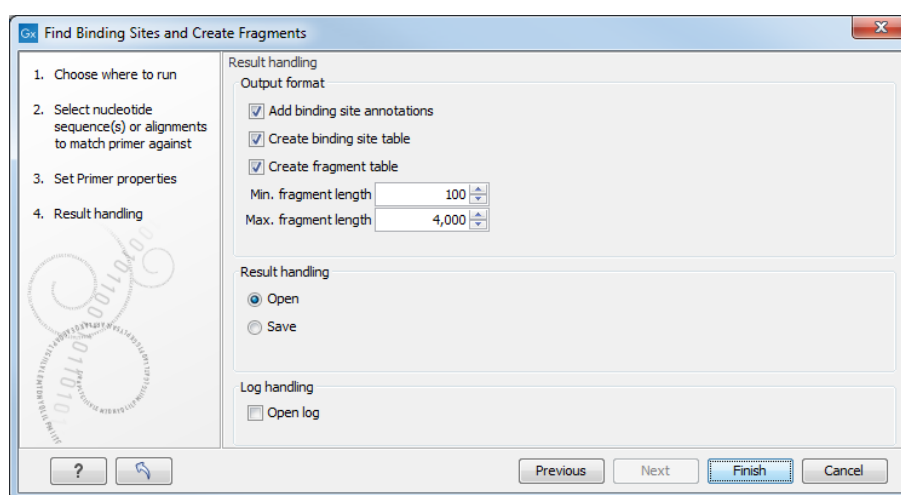


Figure 19.19: Output options include reporting of binding sites and fragments.

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.
- **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 **Finish** to start the tool.

An example of a **binding site annotation** is shown in figure 19.20.

The annotation has the following information:

- **Sequence of the primer.** Positions with mismatches will be in lower-case (see the fourth position in figure 19.20 where the primer has an a and the template sequence has a T).

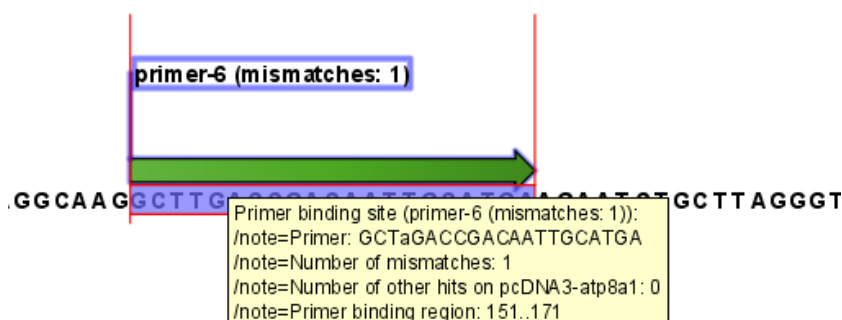


Figure 19.20: Annotation showing a primer match.

- **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 19.21.

Primer name	Primer Sequence	Orientation	Region	Mismatches	Number of other hit...
Primer 1 - HindIII	aaGcttGGTGGGAGGTCTATATAA	fwd	787..810	5	0
Primer 5	ACGGTGGGAGGTCTATATAA	fwd	791..810	0	0
Primer 1	GGTGGGAGGTCTATATAA	fwd	793..810	0	0
Primer 7	GGAACTGAGAAATAGAGGA	rev	complement(1373..1390)	0	0
Primer 6	AGCAGGGCAATAAGAGAA	rev	complement(1412..1430)	0	0

Figure 19.21: 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 19.5.3. You can use this table to browse the binding sites. If you make a split view of the table and the sequence (see section 2.1.6), 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 19.22.

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.

Fwd. name	Rev. name	Fragment length	Region	Other fragments
Primer 1 - HindIII	Primer 7	604	787..1390	0
Primer 5	Primer 7	600	791..1390	0
Primer 1	Primer 7	598	793..1390	0
Primer 1 - HindIII	Primer 6	644	787..1430	0
Primer 5	Primer 6	640	791..1430	0
Primer 1	Primer 6	638	793..1430	0

Figure 19.22: A table showing all possible fragments of the specified size.

You can use this table to browse the fragment regions. If you make a split view of the table and the sequence (see section 2.1.6), 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 19.23.

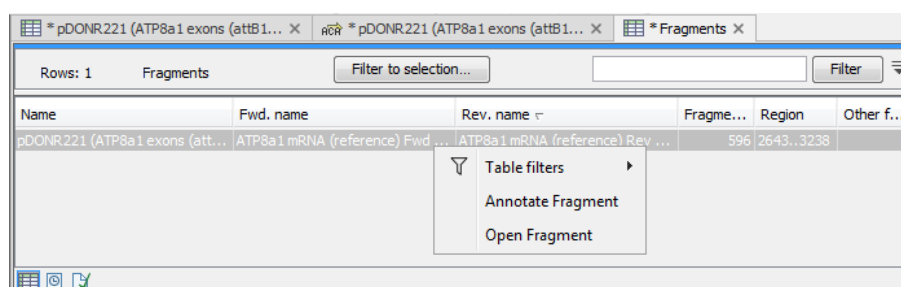


Figure 19.23: Right-clicking a fragment allows you to annotate the region on the input sequence or open the fragment as a new sequence.

This will put a *PCR fragment* annotations on the input sequence covering the region specified in the table. As you can see from figure 19.23, 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 20.3).

19.12 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:

Toolbox | Primers and Probes (📁) | Order Primers (📄)

This opens a dialog where you can choose primers to generate a textual representation of the primers (see figure 19.24).

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. This file can also be saved and exported as a text file.

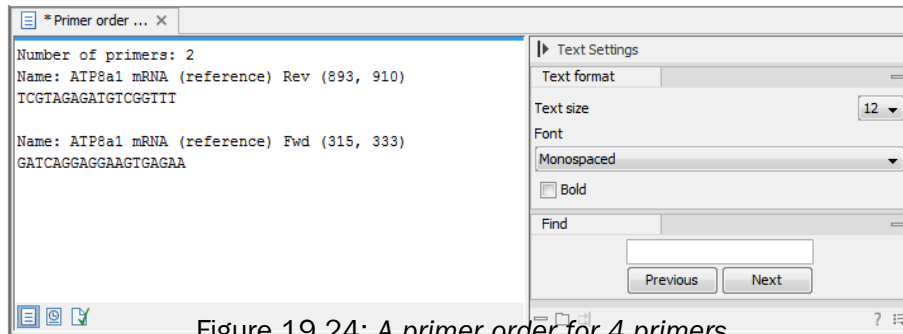


Figure 19.24: A primer order for 4 primers.

Chapter 20

Cloning and restriction sites

Contents

20.1 Restriction site analyses	401
20.1.1 Dynamic restriction sites	402
20.1.2 Restriction Site Analysis	407
20.2 Restriction enzyme lists	410
20.3 Molecular cloning	411
20.3.1 Introduction to the cloning editor	412
20.3.2 The cloning workflow	413
20.3.3 Manual cloning	415
20.3.4 Insert restriction site	420
20.4 Gateway cloning	420
20.4.1 Add attB sites	421
20.4.2 Create entry clones (BP)	424
20.4.3 Create expression clones (LR)	425
20.5 Gel electrophoresis	426
20.5.1 Gel view	427

CLC Main Workbench offers graphically advanced *in silico* cloning and design of vectors, together with restriction enzyme analysis and functionalities for managing lists of restriction enzymes.

20.1 Restriction site analyses

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 is the fastest and easiest way of showing restriction sites.
- In the **Toolbox** you will find the Cloning and Restriction Sites tool that provides more control on the analysis, and gives you more output options such as a table of restriction sites. It also allows you to perform the same restriction map analysis on several sequences in one step.

20.1.1 Dynamic restriction sites

If you open a sequence, a sequence list etc, you will find a **Restriction Sites** section in the Side Panel.

Restriction sites can be shown on the sequence as colored triangles and lines (figure 20.1): check the "Show" option on top of the Restriction sites section, then specify the enzymes that should be displayed.



Figure 20.1: 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 20.2). 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.



Figure 20.2: Restriction site labels shown as flags.

- **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 20.3).

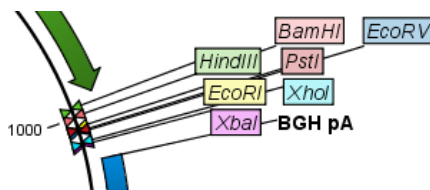


Figure 20.3: Restriction site labels in radial layout.

- **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 20.4).

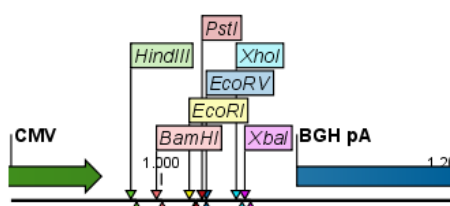


Figure 20.4: Restriction site labels stacked.

Note that in a circular view, the **Stacked** and **Radial** options also affect the layout of annotations. Just above the list of enzymes, three buttons can be used for sorting the list (see figure 20.5).

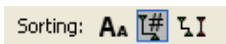


Figure 20.5: Buttons to sort restriction enzymes.

- **Sort enzymes alphabetically (AA).** Clicking this button will sort the list of enzymes alphabetically.
- **Sort enzymes by number of restriction sites (T#).** 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 (LI).** This will divide the enzymes into three groups:
 - Blunt. Enzymes cutting both strands at the same position.
 - 3'. Enzymes producing an overhang at the 3' end.

- 5'. Enzymes producing an overhang at the 5' end.

There is a checkbox for each group which can be used to hide / show all the enzymes in a group.

Manage enzymes

The list of restriction enzymes contains per default some of the most popular enzymes, but you can easily modify this list and add more enzymes by clicking the **Manage enzymes button** found at the bottom of the "Restriction sites" palette of the Side Panel.

This will open the dialog shown in figure 20.6.

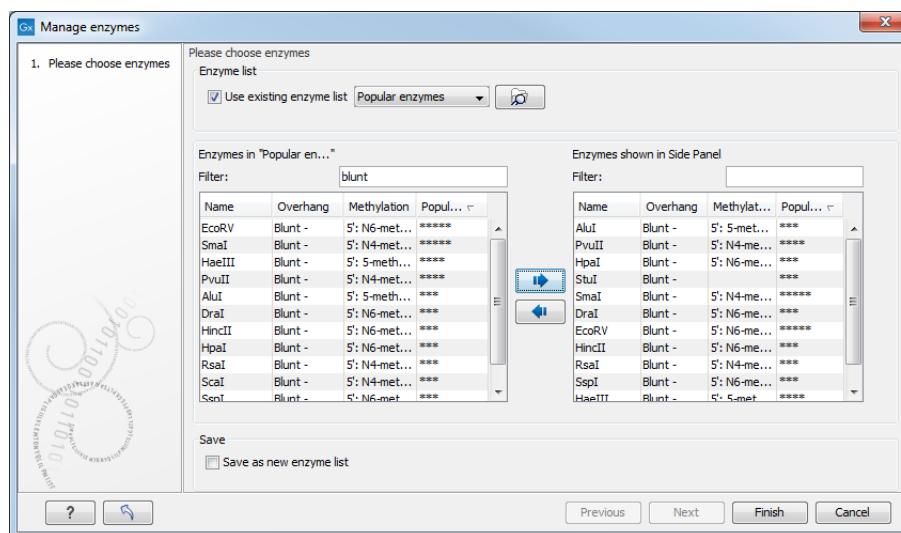


Figure 20.6: Adding or removing enzymes from the Side Panel.

At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. A list of popular enzymes is available in the Example Data folder you can download from the Help menu.

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 a specific enzyme list, this panel shows all the enzymes available.
- 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 (➡).

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 a 3' overhang for example.

When looking for a specific enzyme, it is easier to use the Filter. You can type HindIII or blunt into the filter, and the list of enzymes will shrink automatically to only include respectively only the HindIII enzyme, or all enzymes producing a blunt cut.

If you need more detailed information and filtering of the enzymes, you can hover your mouse on an enzyme (see figure 20.7). You can also open a view of an enzyme list saved in the Navigation Area.

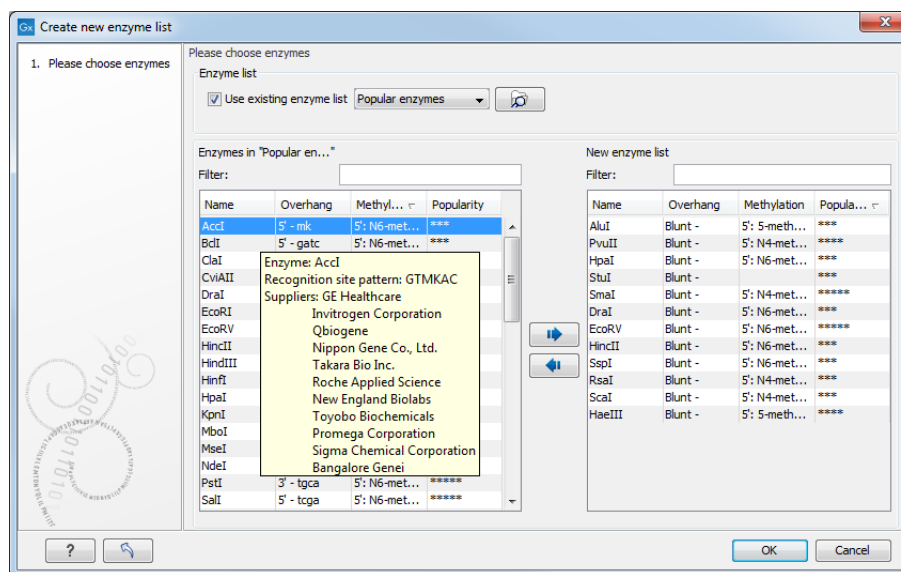


Figure 20.7: Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.

At the bottom of the dialog, you can select to save the updated list of enzymes as a new file. 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 4.6) for future use.

Show enzymes cutting inside/outside selection

In cases where you have a selection on a sequence, and you wish to find enzymes cutting within the selection but not outside, right-click the selection and choose the option **Show Enzymes Cutting Inside/Outside Selection** (🔍).

This will open a wizard where you can specify which enzymes should initially be considered (see section 20.1.1). You can for example select all the enzymes from a custom made list that correspond to all the enzymes that are already available in your lab.

In the following step (figure 20.8), you can define the terms of your search.

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.
- **Outside selection.** Specify how many times you wish the enzyme to cut outside the selection (i.e. the rest of the sequence).

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

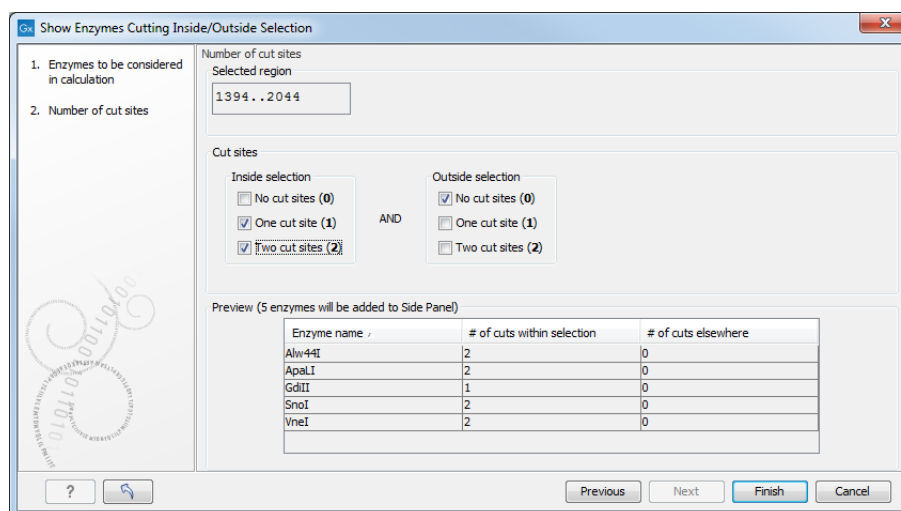


Figure 20.8: Deciding number of cut sites inside and outside the selection.

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

A third way of adding enzymes to the Side Panel and thereby displaying them on the sequence is based on the overhang produced by cutting with an enzyme. Right-click on a restriction site and choose to **Show Enzymes with Compatible Ends** (⌘) to find enzymes producing a compatible overhang (figure 20.9).

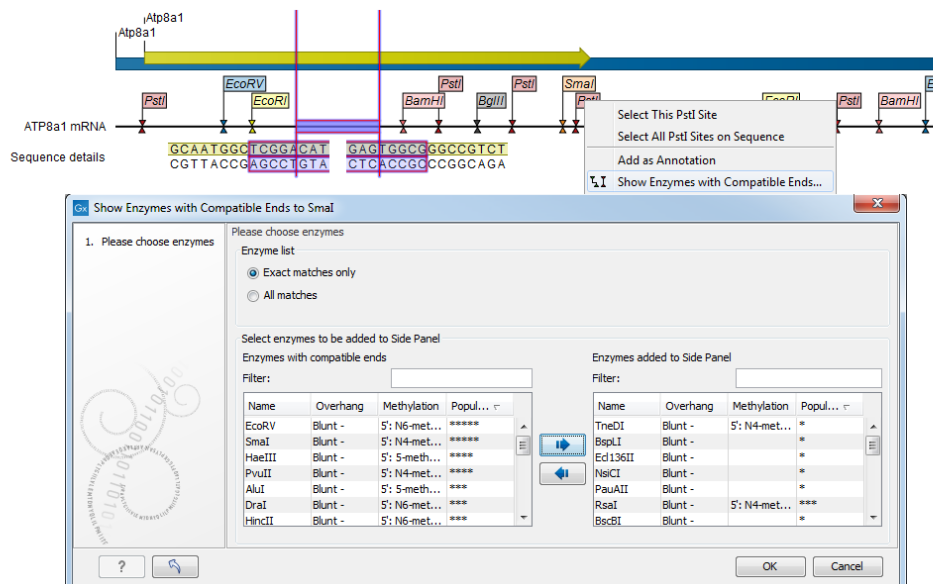


Figure 20.9: Enzymes with compatible ends.

At the top you can choose whether the enzymes considered should have an exact match or not. We recommend trying **Exact match** first, and use **All matches** as an alternative if a satisfactory

result cannot be achieved. Indeed, 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.

Use the arrows between the two panels to select enzymes which will be displayed on the sequence and added to the Side Panel.

At the bottom of the dialog, the list of enzymes producing compatible overhangs is shown.

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.

20.1.2 Restriction Site Analysis

Besides the dynamic restriction sites, you can do a more elaborate restriction map analysis with more output format using the Toolbox:

Toolbox | Cloning and Restriction Sites (🔗) | Restriction Site Analysis (✂️)

You first specify which sequence should be used for the analysis. Then define which enzymes to use as basis for finding restriction sites on the sequence (see section 20.1.1).

In the next dialog, you can use the checkboxes so that the output of the restriction map analysis only include restriction enzymes which cut the sequence a specific number of times (figure 20.10).

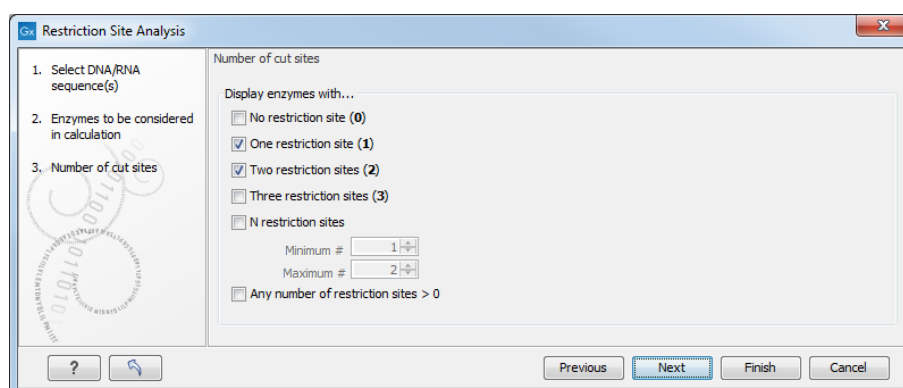


Figure 20.10: Selecting number of cut sites.

The default setting is to include the enzymes which cut the sequence one or two times, but you can use the checkboxes to perform very specific searches for restriction sites, for example to find enzymes which do not cut the sequence, or enzymes cutting exactly twice.

The Result handling dialog (figure 20.11) 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 20.12) and save the annotations for later use.

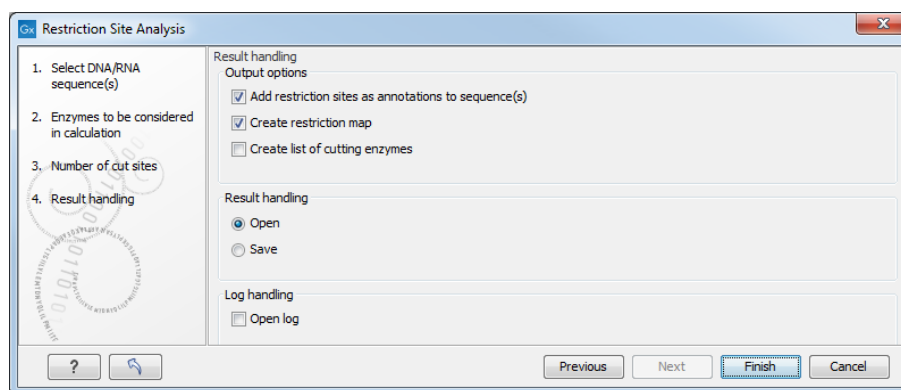


Figure 20.11: Choosing to add restriction sites as annotations or creating a restriction map.



Figure 20.12: The result of the restriction analysis shown as annotations.

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 20.13. 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.

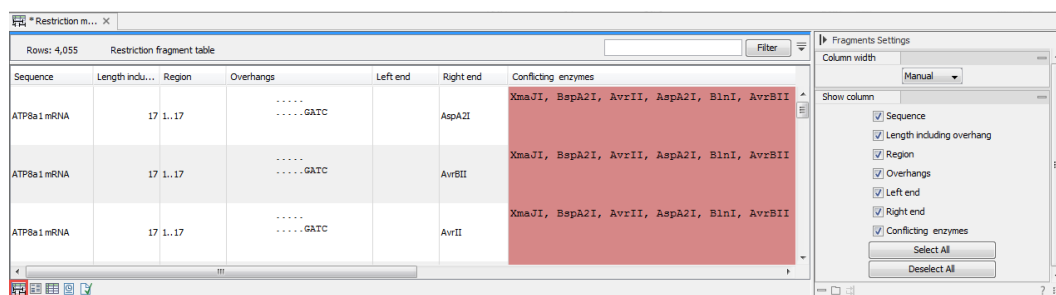
Sequence	Name	Pattern	Length	Overhang	Number of c...	Cut position(s)
ATP8a1 mRNA	AagI	atcgat	6 5'		1	3895
ATP8a1 mRNA	AarI	cacctgc	7 5'		1	212
ATP8a1 mRNA	AasI	gaacnnmngtc	12 3'		1	6531
ATP8a1 mRNA	AatI	agcctc	6 Blunt		2	3679, 6393
ATP8a1 mRNA	AatII	gacgtc	6 3'		1	2593
ATP8a1 mRNA	AbaI	tgatca	6 5'		2	403, 4981
ATP8a1 mRNA	AbeI	cctcagc	7 5'		2	352, [1986]
ATP8a1 mRNA	Acc113I	agtact	6 Blunt		1	386
ATP8a1 mRNA	Acc65I	ggtacc	6 5'		1	1204
ATP8a1 mRNA	AccEBI	ggatcc	6 5'		2	2221, 5899
ATP8a1 mRNA	AccIII	tcggga	6 5'		2	3291, 4074
ATP8a1 mRNA	AdI	aacgtt	6 5'		2	4495, 6208
ATP8a1 mRNA	AcrII	ggtnacc	7 5'		1	5633
ATP8a1 mRNA	AcvI	cacgtg	6 Blunt		1	3530

Figure 20.13: The result of the restriction analysis shown as a table of restriction sites.

Each row in the table represents a restriction enzyme. The following information is available for each enzyme:

- **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.
- **Length.** the restriction site length.
- **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's recognition sequence is on the negative strand, the cut position is put in brackets.
 - * **()** Some enzymes cut the sequence twice for each recognition site, and in this case the two cut positions are surrounded by parentheses.
- As a **table of fragments** which shows the sequence fragments that would be the result of cutting the sequence with the selected enzymes (see figure 20.14). Click the Fragments button () at the bottom of the view.



Sequence	Length including overhang	Region	Overhangs	Left end	Right end	Conflicting enzymes
ATP8a1 mRNA	17	1..17GATC		AspA2I	XmaJI, BspA2I, AvrII, AspA2I, BlnI, AvrBII
ATP8a1 mRNA	17	1..17GATC		AvrBII	XmaJI, BspA2I, AvrII, AspA2I, BlnI, AvrBII
ATP8a1 mRNA	17	1..17GATC		AvrII	XmaJI, BspA2I, AvrII, AspA2I, BlnI, AvrBII

Figure 20.14: The result of the restriction analysis shown as table of fragments.

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. Furthermore, if this is the case, you will see the names of the other enzymes in the **Conflicting Enzymes** column.

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 including overhang.** 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.
- **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.

- As a **virtual gel** simulation which shows the fragments as bands on a gel (see figure 20.40). For more information about gel electrophoresis, see section 20.5.

20.2 Restriction enzyme lists

CLC Main Workbench includes all the restriction enzymes available in the **REBASE** database, with methylation shown as performed by the cognate methylase rather than by Dam/Dcm. If you want to customize the enzyme database for your installation, see section D. 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 for example all enzymes available in the laboratory freezer, or all enzymes used to create a given restriction map or all enzymes that are available from the preferred vendor.

In the Example data (import in your Navigation Area using the Help menu), 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*.

Create enzyme list *CLC Main Workbench* uses enzymes from the **REBASE** restriction enzyme database at <http://rebase.neb.com>. If you want to customize the enzyme database for your installation, see section D.

To create an enzyme list of a subset of these enzymes:

File | New | Enzyme list (🧬)

This opens the dialog shown in figure 20.15

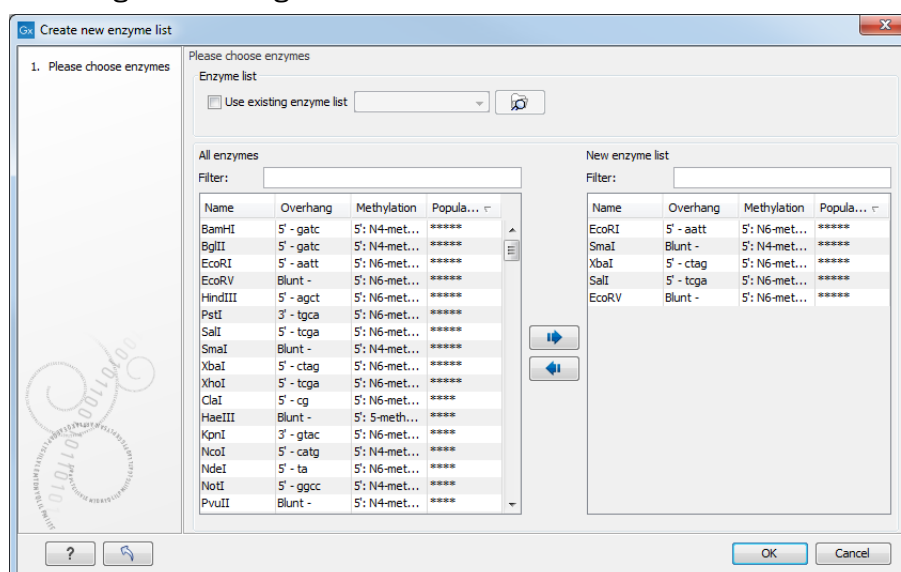


Figure 20.15: Choosing enzymes for the new enzyme list.

Choose which enzyme you want to include in the new enzyme list (see section 20.1.1), and click **Finish** to open the enzyme list.

View and modify enzyme list An enzyme list is shown in figure 20.16. It can be sorted by clicking the columns, and you can use the filter at the top right corner to search for specific

enzymes, recognition sequences etc.

Name	Recognition sequence	Length	Overhang	Suppliers
PshAI	gacnngtc	10	Blunt	GE Healthcare; Takara Bio Inc.; New England Biolabs
ClaI	atcgat	6	5' - cg	GE Healthcare; Invitrogen Corporation; American Allied Biochemical, Inc.; Takara Bio Inc.; Roche Applied Science
Uba153AI	cagctg	6	Blunt	
AsiSI	gcgatcgc	8	3' - at	New England Biolabs
MlyI	ccwgg	5	5' - w	GE Healthcare; Fermentas International Inc.; Qbiogene; Takara Bio Inc.; Roche Applied Science; Takara Bio Inc.
Mly113I	Name of the enzyme	6	5' - cg	SibEnzyme Ltd.
Bce243I	gatc	4	5' - gatc	
Bsp2095I	gatc	4	5' - gatc	
BetI	wccggw	6	5' - ccgg	
BspLS2I	gdgchc	6	3' - dgch	
MreI	cgccggcg	8	5' - ccgg	
Bsp119I	ttcgaa	6	5' - cg	Fermentas International Inc.
BmcAI	agtact	6	Blunt	Vivantis Technologies
Sru30DI	aggcct	6	Blunt	
BstBS32I	gaagac	6	5' - <NA>	
Hpy178III	tcnnga	6	5' - nn	
BmuI	actggg	6	3' - <NA>	SibEnzyme Ltd.
BspT104I	ttcgaa	6	5' - cg	Takara Bio Inc.
BstDEI	ctnag	5	5' - tna	SibEnzyme Ltd.; Vivantis Technologies
NciI	gcgccgc	8	5' - ggcc	GE Healthcare; Invitrogen Corporation; Minotech Biotechnology; Fermentas International Inc.; Qbiogene
SgrBI	ccgcg	6	3' - gc	Minotech Biotechnology
AccB2I	rgcgcy	6	3' - gcgc	
Bbv12I	gwgcw	6	3' - wgcw	SibEnzyme Ltd.; Vivantis Technologies
BavAI	cagctg	6	Blunt	
RohI	ancc	4	Blunt	

Figure 20.16: An enzyme list.

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 20.15 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 on the selection and choose to **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. For example, 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.

20.3 Molecular cloning

The *in silico* cloning process in *CLC Main Workbench* begins with the Cloning tool:

Cloning and Restriction Sites (🔗) | Cloning (🔗)

This will open a dialog where you can select both the sequences containing the fragments you want to clone, as well as the one to be used as vector (figure 20.17).

CLC Main Workbench will now create a sequence list of the selected fragments and vector sequences. For cloning work, open the sequence list and switch to the **Cloning** (🔗) editor at the bottom of the view (figure 20.18).

If you later in the process need additional sequences, right-click anywhere on the empty white area of the view and choose to "Add Sequences".

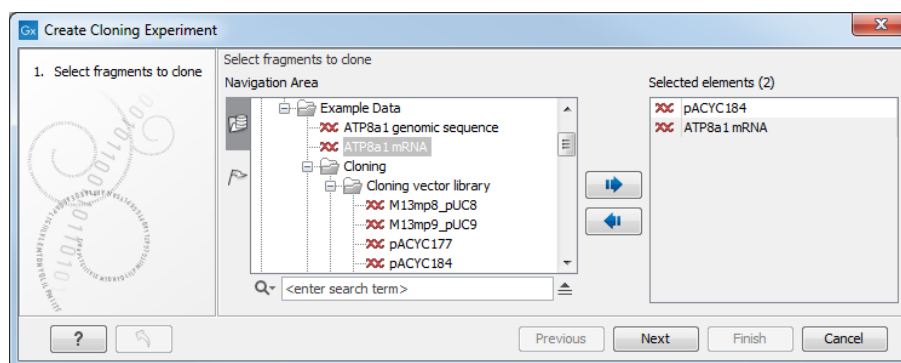


Figure 20.17: Selecting the sequences containing the fragments you want to clone and the vector.

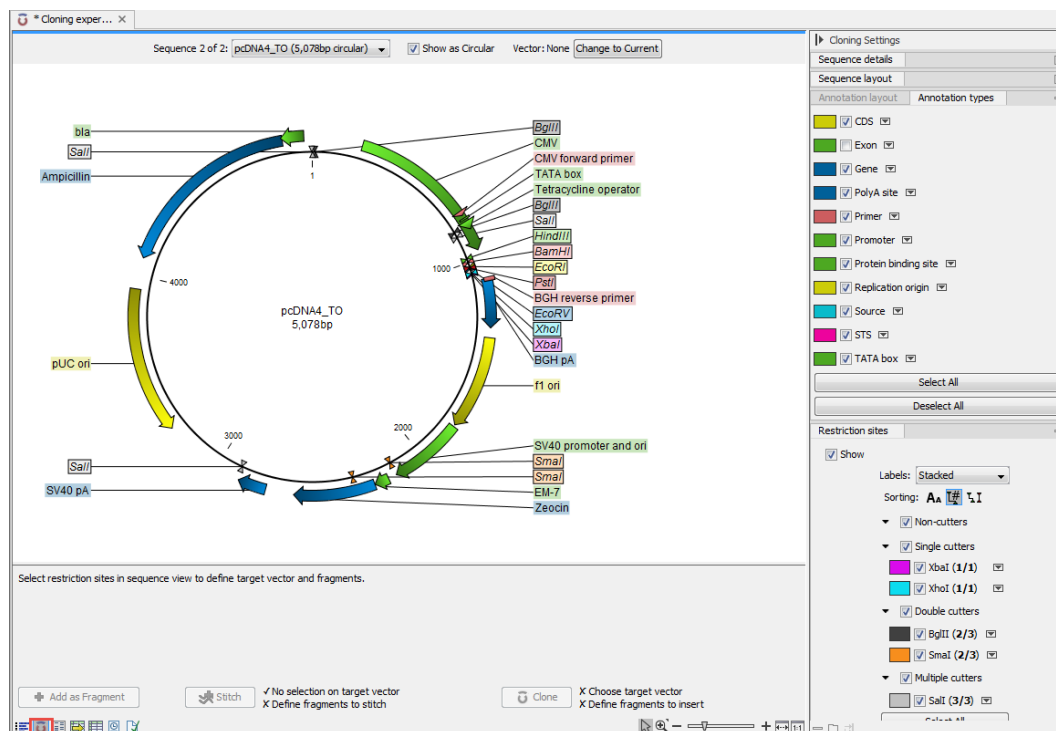


Figure 20.18: Cloning editor view of the sequence list. Choose which sequence to display from the drop down menu.

20.3.1 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 (section 11.1). In particular, this means that 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. On the right-hand side, you can select a vector: the button is by default set to **Change to Current**. Click on it 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.

- At the bottom, there is a panel where the selection of fragments and target vector is performed.

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:

- **The cloning mode**, 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.
- **The stitch mode**, when the user 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.

20.3.2 The cloning workflow


The *cloning workflow* is designed to support restriction cloning workflows through the following steps:


1. Define one or more 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 20.1.1). 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. If you just wish to remove the selection of one of the sites, right-click the site on the sequence and choose **De-select This ... Site**.

When this is done, the panel is updated to reflect the selections (see figure 20.19).

In this example you can see that there are now three fragments that can be used for cloning listed in the panel below the view. 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 **Add as Fragment** ().

At any time, the selection of cut sites can be cleared by clicking the **Remove** () icon to the right of the target vector selections.

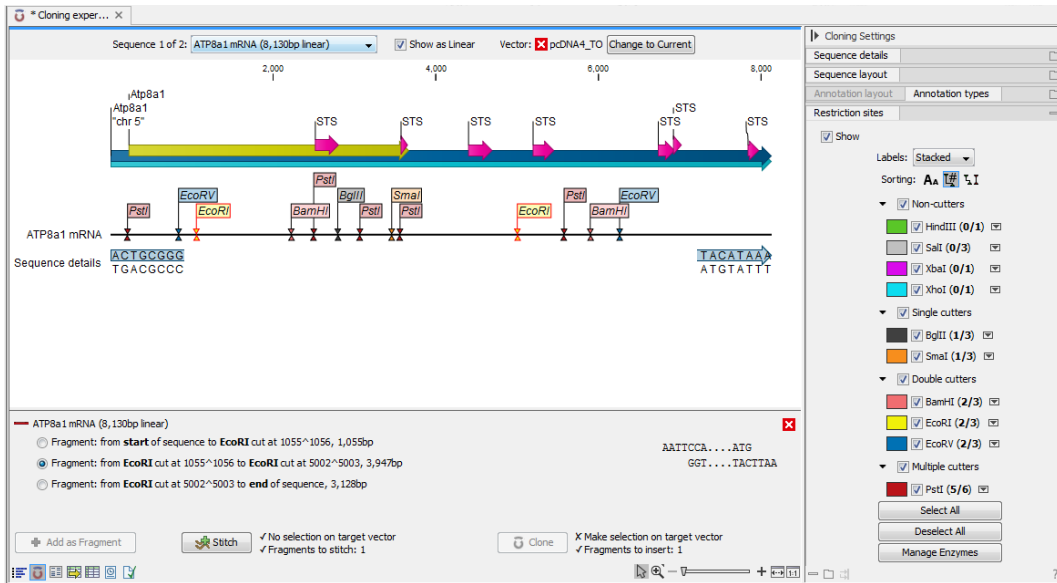


Figure 20.19: EcoRI cut sites selected to cut out fragment.

2. Defining target vector

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 (figure 20.20).

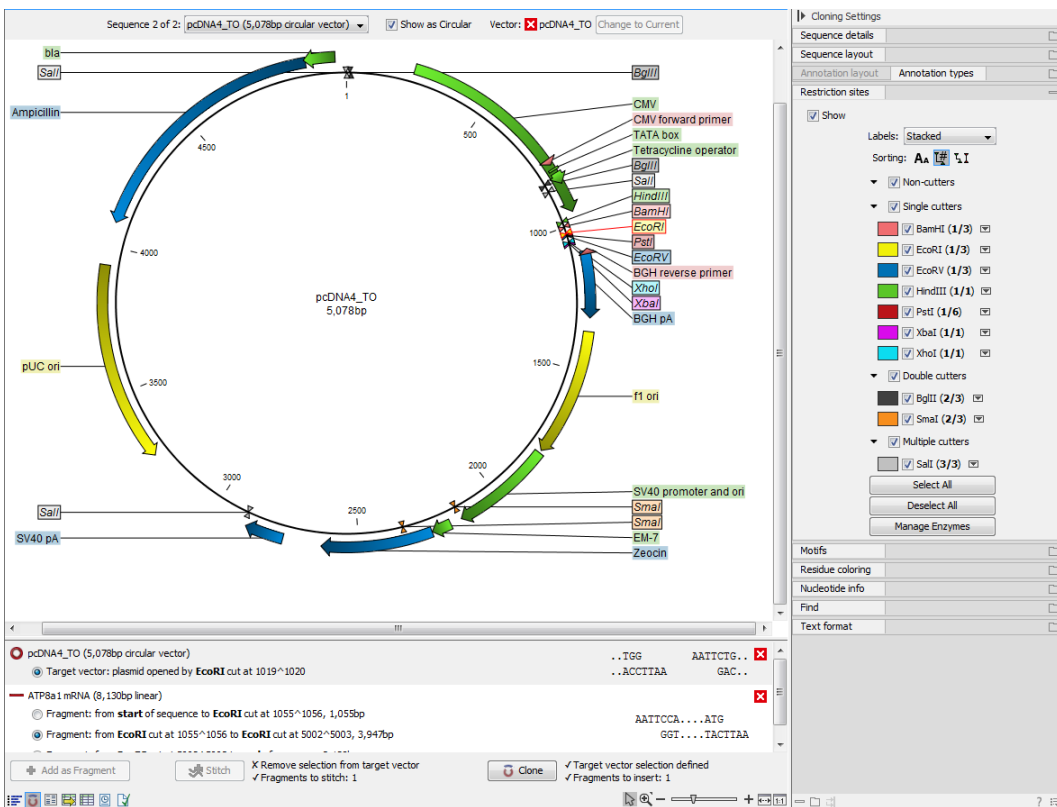


Figure 20.20: EcoRI site used to open the vector. Note that the "Cloning" button has now been enabled as both criteria ("Target vector selection defined" and "Fragments to insert:...") have been defined.

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). At any time, the selection of cut sites can be cleared by clicking the **Remove** (✖) icon to the right of the target vector selections.

3. Perform cloning

Once both fragments and vector are selected, click **Clone** (🔄). This will display a dialog to adapt overhangs and change orientation as shown in figure 20.21)

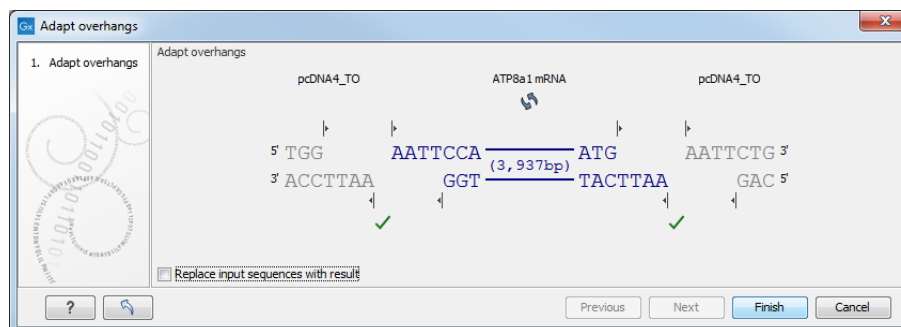


Figure 20.21: Showing the insertion point of the vector.

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 will not be able to click **Finish**. But you can blunt end or fill in the overhangs using the **drag handles** (⏏) until 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 (➡)/(⬅).

Per default, the construct will be opened in a new view and can be saved separately. But selecting the option **Replace input sequences with result** will add the construct to the input sequence list and delete the original fragment and vector sequences.

Note that 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.

20.3.3 Manual cloning

If you wish to use the manual way of cloning, you still create a sequence list with the Cloning tool, but can skip the "Perform cloning" step of the cloning workflow explained above in section 20.3.2. Instead, all manipulations of sequences are done manually, using right-click menus. These menus have two different appearances depending on where you click, as visualized in figure 20.22.

Manipulate the whole sequence

Right-click the sequence label to the left to see the menu shown in figure 20.23.

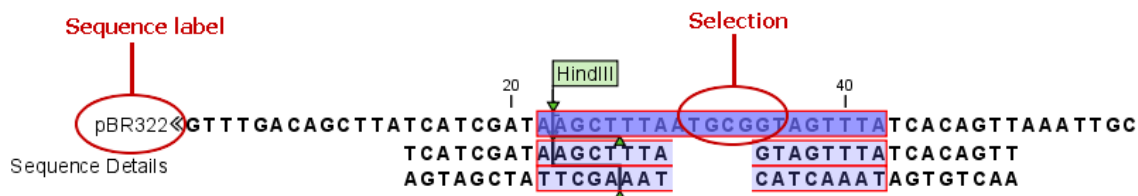


Figure 20.22: The red circles mark the two places you can use for manipulating the sequences.

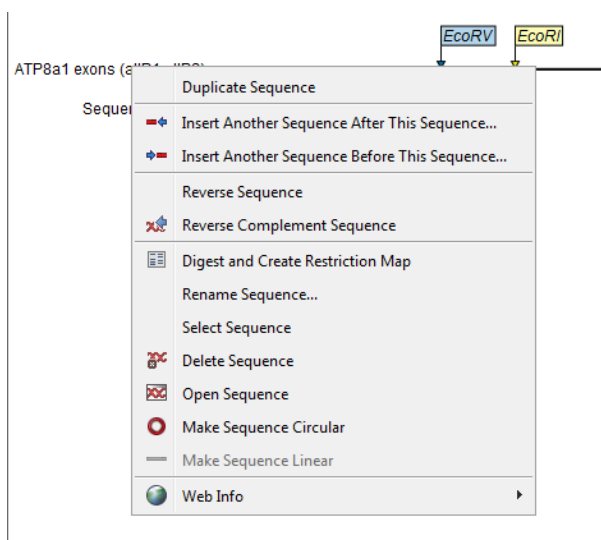


Figure 20.23: Right click on the sequence in the cloning view.

- **Duplicate sequence.** Adds a duplicate of the selected sequence to the sequence list accessible from the drop down menu on top of the Cloning view.
- **Insert sequence after this sequence** (➡➡). The sequence to be inserted can be selected from the sequence list via the drop down menu on top of the Cloning view. 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.
- **Insert sequence before this sequence** (➡➡). The sequence to be inserted can be selected from the sequence list via the drop down menu on top of the Cloning view. 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.
- **Reverse sequence.** Reverses 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 of a sequence.
- **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 and Create Restriction Map** (📄). See section 20.5
- **Rename sequence.** Renames the sequence.
- **Select sequence.** Selects the entire sequence.

- **Delete sequence** (✂). Deletes the given sequence from the cloning editor.
- **Open sequence** (📄). Opens the selected sequence in a normal sequence view.
- **Make sequence circular** (🔄). Converts 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** (—). Converts a sequence from a circular to a linear form, removing the << and >> at the ends.

Manipulate parts of the sequence

Right-click on a selected region of the sequence to see the menu shown in figure 20.24.

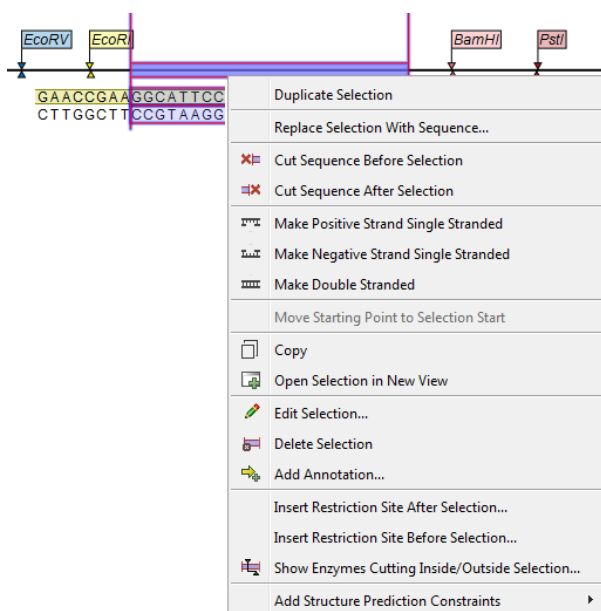











Figure 20.24: Right click on a sequence selection in the cloning view.

- **Duplicate Selection.** If a selection on the sequence is duplicated, the selected region will be added as a new sequence to the cloning editor. The new sequence name representing the length of the fragment. When double-clicking on a sequence, the region between the two closest restriction sites is automatically selected.
- **Replace Selection with sequence.** Replaces the selected region with a sequence selected from the drop down menu listing all sequences in the cloning editor.
- **Cut Sequence Before Selection** (✂). Cleaves the sequence before the selection and will result in two smaller fragments.
- **Cut Sequence After Selection** (✂). Cleaves the sequence after the selection and will result in two smaller fragments.

- **Make Positive Strand Single Stranded** (). Makes the positive strand of the selected region single stranded.
- **Make Negative Strand Single Stranded** (). Makes 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** (). Copies the selected region to the clipboard, which will enable it for use in other programs.
- **Open Selection in New View** (). Opens the selected region in the normal sequence view.
- **Edit Selection** (). Opens a dialog box in which is it possible to edit the selected residues.
- **Delete Selection** (). Deletes the selected region of the sequence.
- **Add Annotation** (). Opens the **Add annotation** dialog box.
- **Insert Restriction Sites After/Before Selection**. Shows a dialog where you can choose from a list restriction enzymes (see section 20.3.4).
- **Show Enzymes Cutting Inside/Outside Selection** (). Adds enzymes cutting this selection to the Side Panel.
- **Add Structure Prediction Constraints**. This is relevant for RNA secondary structure prediction:
 - **Force Stem Here** is activated after choosing 2 regions of equal length on the sequence. It will add an annotation labeled "Forced Stem" and will force the algorithm to compute minimum free energy and structure with a stem in the selected region.
 - **Prohibit Stem Here** is activated after choosing 2 regions of equal length on the sequence. It will add an annotation labeled "Prohibited Stem" to the sequence and will force the algorithm to compute minimum free energy and structure without a stem in the selected region.
 - **Prohibit From Forming Base Pairs** will add an annotation labeled "No base pairs" to the sequence, and will force the algorithm to compute minimum free energy and structure without a base pair containing any residues in the selected region.

Insert one sequence into another

Sequences can be inserted into each other in various ways as described in the lists above. When you choose to insert one sequence into another, you will be presented with a dialog where all sequences in the sequence list are present (see figure 20.25).

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 20.3.2.

Furthermore, the list 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).

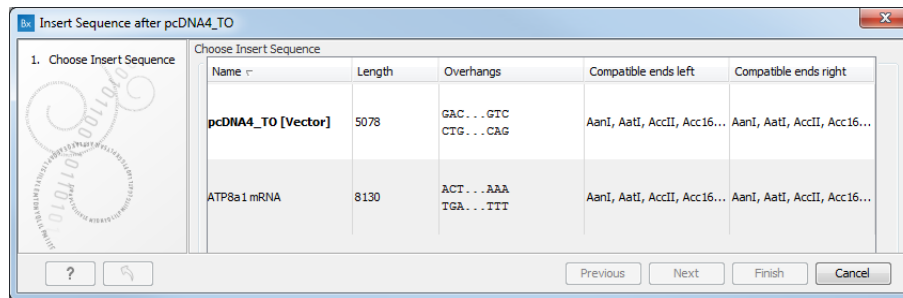


Figure 20.25: Select a sequence for insertion.

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** to adapt insert sequence to vector dialog (figure 20.26).

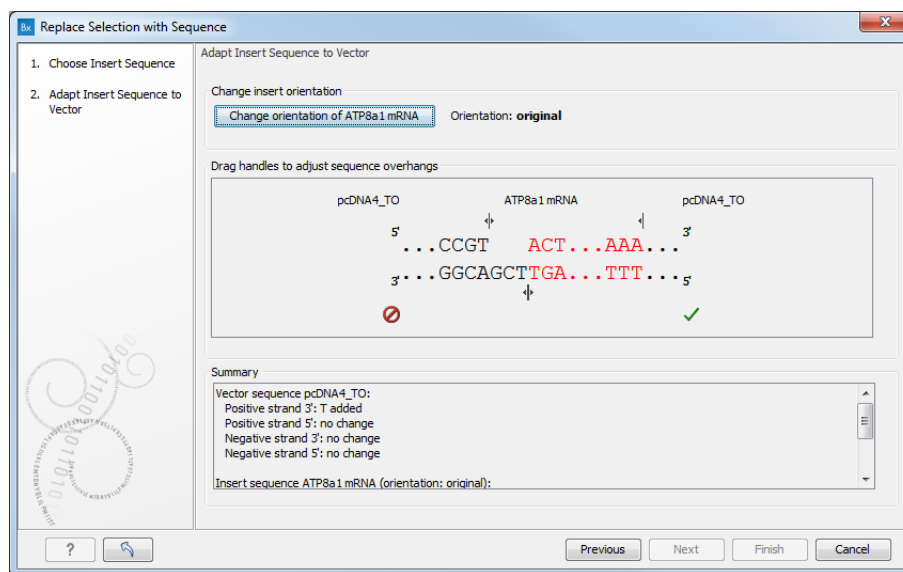


Figure 20.26: Drag the handles to adjust overhangs.

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** (|) until it does (✓).

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 (📄) of the cloning experiment.

When you click **Finish**, the sequence is inserted and highlighted by being selected.

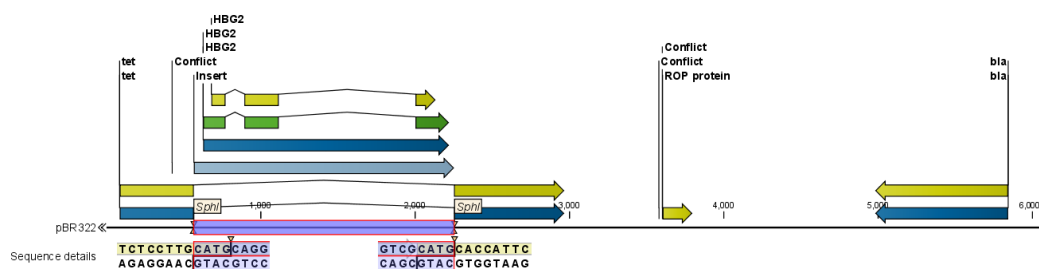


Figure 20.27: One sequence is now inserted into the cloning vector. The sequence inserted is automatically selected.

20.3.4 Insert restriction site

If you right-click on a selected region of a sequence, 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 20.28

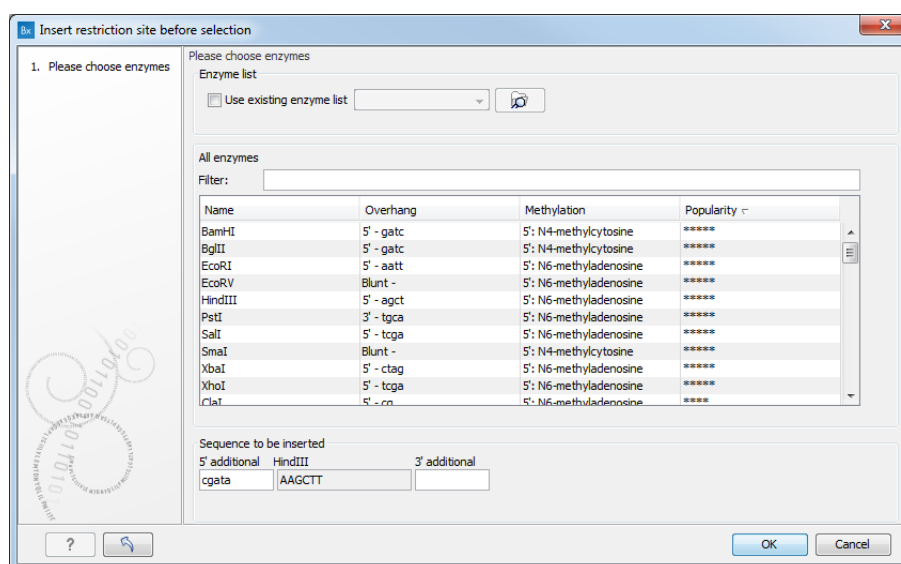


Figure 20.28: Inserting a restriction site and potentially a recognition sequence.

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, this can be typed into the text fields adjacent to the recognition sequence.

Click **OK** will insert the restriction site and the tag(s) 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.

20.4 Gateway cloning

CLC Main Workbench offers tools to perform *in silico* Gateway cloning¹, including Multi-site Gateway cloning.

The three tools for doing Gateway cloning in the CLC Main Workbench mimic the procedure

¹Gateway is a registered trademark of Invitrogen Corporation

followed in the lab:

- 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.thermofisher.com/us/en/home/life-science/cloning/gateway-cloning/gateway-technology.html>. To perform these analyses in *CLC Main Workbench*, you need to import donor and expression vectors. These can be found on the Thermo Fisher Scientific's website: find the relevant vector sequences, copy them, and paste them in the field that opens when you choose **New | Sequence** in the workbench. Fill in additional information appropriately (enter a "Name", check the "Circular" option) and save the sequences in the Navigation Area.

20.4.1 Add attB sites

The first step in the Gateway cloning process is to amplify the target sequence with primers including so-called attB sites:

Toolbox | Cloning and Restriction Sites (🔧) | Gateway Cloning (📁) | Add attB Sites (🔗)

This will open a dialog where you can select one or more sequences. Note that if your fragment is part of a longer sequence, you will need to extract it prior to starting the tool: select the relevant region (or an annotation) of the original sequence, right-click the selection and choose to **Open Annotation in New View**. **Save** (📁) the new sequence in the Navigation Area.

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 20.29.

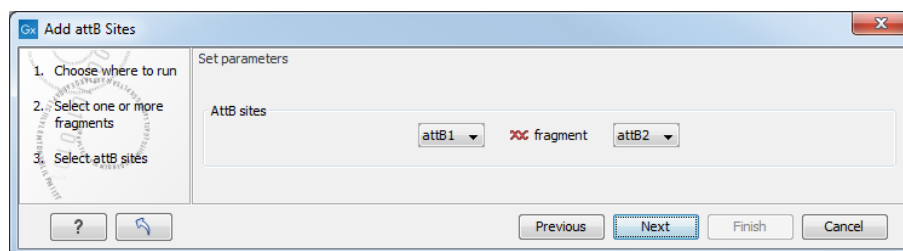


Figure 20.29: Selecting which attB sites to add.

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.

Next, you are given the option 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.

You can manually type or paste in a sequence of your choice, but it is also possible to click in the text field and press **Shift + F1 (Shift + Fn + F1 on Mac)** to show some of the most common additions (see figure 20.30). Use the up and down arrow keys to select a tag and press **Enter**. To learn how to modify the default list of primer additions, see section 20.4.1.

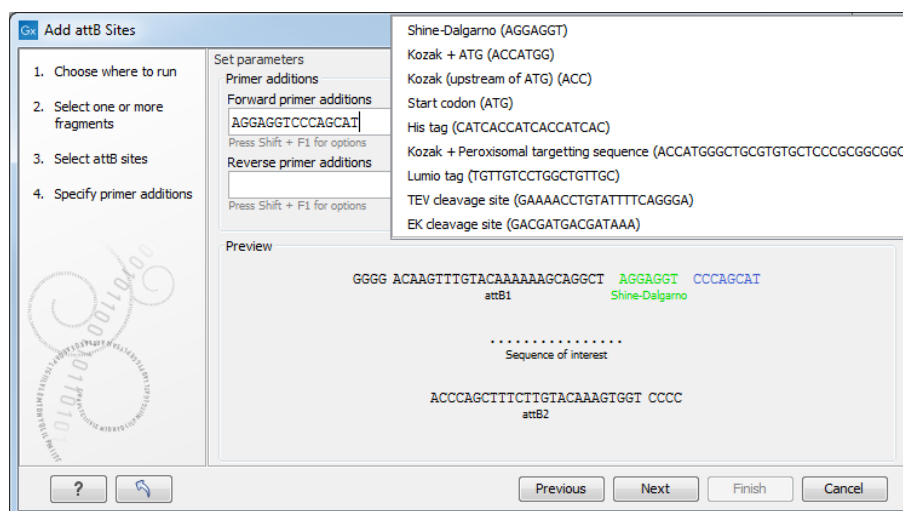


Figure 20.30: Primer additions 5' of the template-specific part of the primer where a Shine-Dalgarno site has been added between the attB site and the gene of interest.

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. 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.

In the next step, specify the length of the template-specific part of the primers as shown in figure 20.31.

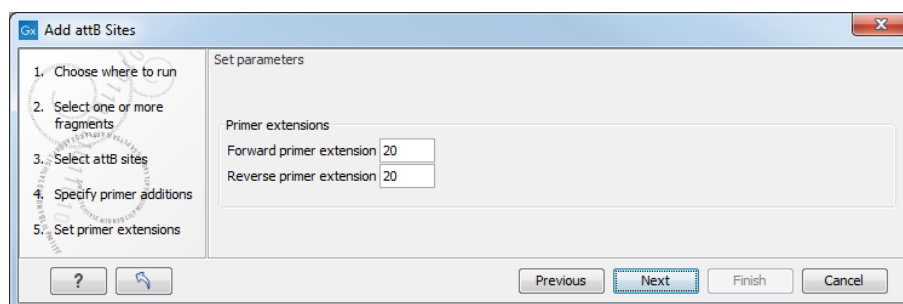


Figure 20.31: Specifying the length of the template-specific part of the primers.

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. You can also choose to get a list of primers in the Result handling dialog (see figure 20.32).

The attB sites, the primer additions and the primer regions are annotated in the final result as shown in figure 20.33 (you may need to switch on the relevant annotation types to show the

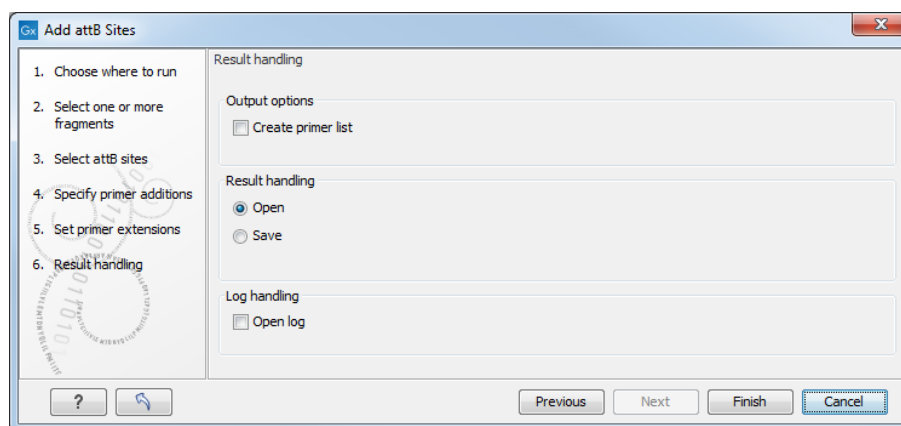


Figure 20.32: Besides the main output which is a copy of the input sequence(s) now including attB sites and primer additions, you can get a list of primers as output.

sites and primer additions).

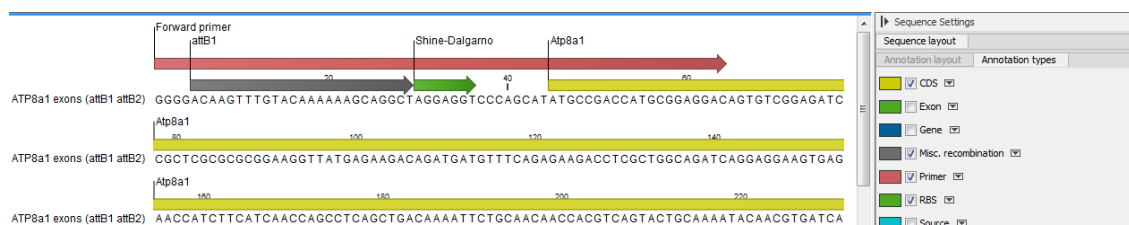


Figure 20.33: the attB site plus the Shine-Dalgarno primer addition is annotated.

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 workflow (see section 20.4.2).

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 20.30 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 | Data

In the table **Multisite Gateway Cloning primer additions** (see figure 20.34), select which primer addition options you want to add to forward or reverse primers. You can edit 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.

Each element in the list has the following information:

Name 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, defined on the sense strand (although the reverse primer would be reverse complement).

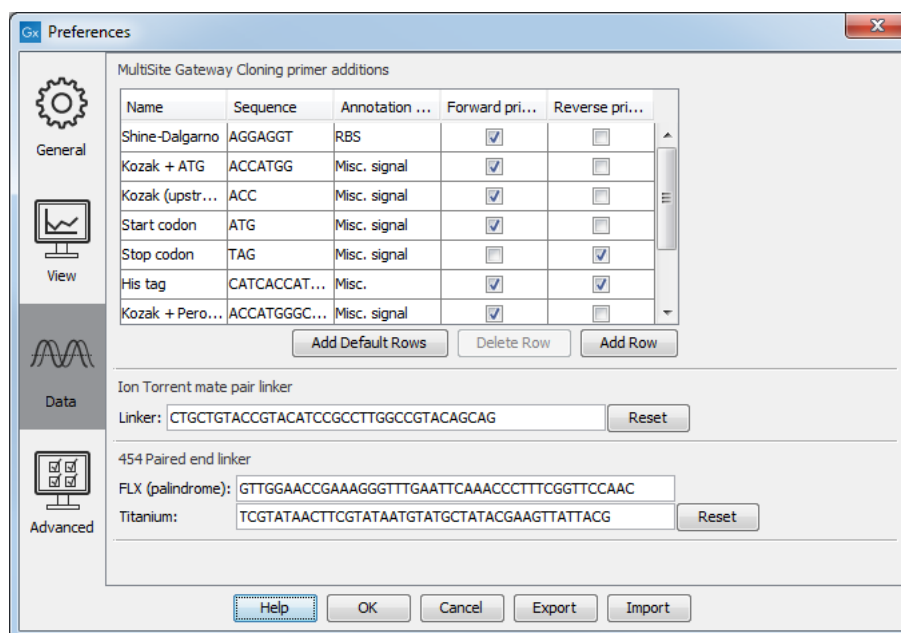


Figure 20.34: Configuring the list of primer additions available when adding attB sites.

Annotation type The annotation type of the primer 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.

20.4.2 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. Before proceeding to this step, make sure that the sequence of the destination vector was saved in the Navigation Area: find the relevant vector sequence on the Thermo Fisher Scientific's website, copy it, and paste it in the field that opens when you choose **New | Sequence** in the workbench. Fill in additional information appropriately (enter a "Name", check the "Circular" option) and save the sequence in the Navigation Area.

Toolbox | Cloning and Restriction Sites (🔧) | Gateway Cloning (📁) | Create Entry Clone (🔄)

In the first wizard window, select one or more sequences to be recombined into your donor vector. Note that the sequences you select should be flanked with attB sites (see section 20.4.1). You can select more than one sequence as input, and the corresponding number of entry clones will be created.

In the following dialog (figure 20.35), you can specify a donor vector.

Once the vector is selected, a preview of the fragments selected and the attB sites that they contain is shown. 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. Also note that the workbench looks for the attP sites (see how to change the definition of sites in appendix E), but it does not

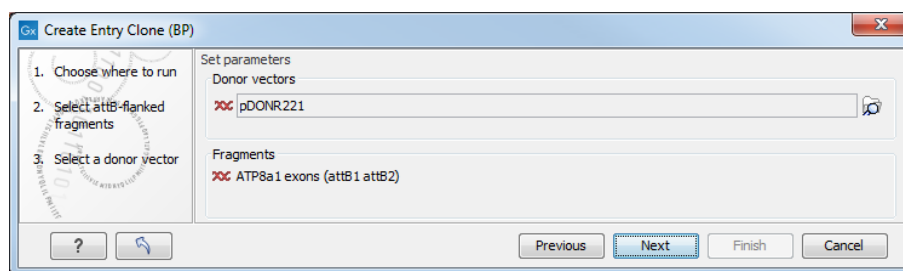


Figure 20.35: Selecting one or more donor vectors.

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.

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 20.36.

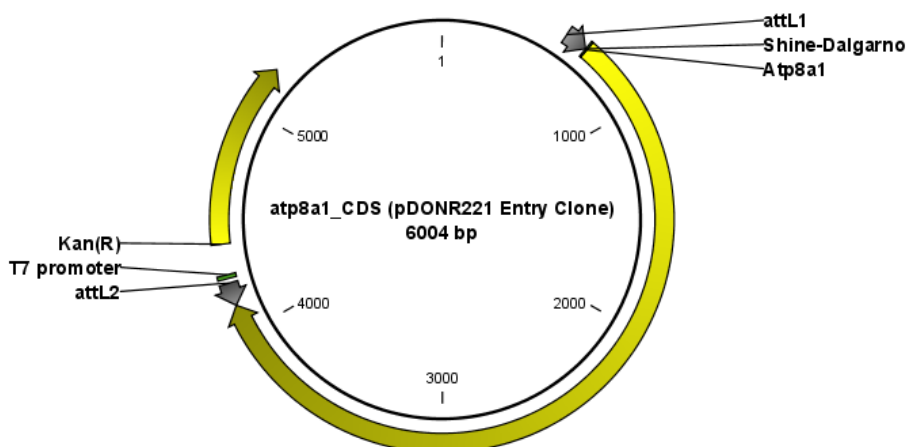


Figure 20.36: The resulting entry vector opened in a circular view.

Note that the bi-product of the recombination is not part of the output.

20.4.3 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. Before proceeding to this step, make sure that the sequence of the destination vector was saved in the Navigation Area: find the relevant vector sequence on the Thermo Fisher Scientific's website, copy it, and paste it in in the field that opens when you choose **New | Sequence** in the workbench. Fill in additional information appropriately (enter a "Name", check the "Circular" option) and save the sequence in the Navigation Area.

Note also that for a destination vector to be recognized, it must contain appropriate att sites and the *ccdB* gene. This gene must be present either as a 'ccdB' annotation, or as the exact sequence:

```
ATGCAGTTTAAGGTTTACACCTATAAAAGAGAGACCGTTATCGTCTGTTTGTGGATGTACAGAGTGATATT
ATTGACACGCCCGGGCGACGGATGGTATCCCCCTGGCCAGTGCACGTCTGCTGTCAGATAAAGTCTCC
CGTGAAC TTTACCCGGTGGTGCATATCGGGGATGAAAGCTGGCGCATGATGACCACCGATATGGCCAGT
GTGCCGGTCTCCGTTATCGGGGAAGAAGTGGCTGATCTCAGCCACCGCGAAAATGACATCAAAAACGCC
```

ATTAACCTGATGTTCTGGGGAATATAA

If the *ccdB* gene is not present or if the sequence is not identical to the above, a solution is to simply add a 'ccdB' annotation. Select part of the vector sequence, right-click and choose 'Add Annotation'. Name the annotation 'ccdB'.

You can now start the tool:

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

In the first step, select one or more entry clones (see how to create an entry clone in section 20.4.2). If you wish to perform separate LR reactions with multiple entry clones, you should run the **Create Expression Clone** in batch mode (see section 8.3).

In the second step, select the destination vector that was previously saved in the Navigation Area (fig 20.37).

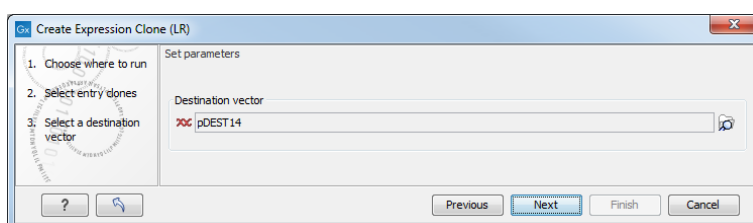


Figure 20.37: Selecting one or more destination vectors.

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 E), but it does not check that they correspond to the attL sites of the selected fragments. If the right combination of attL and attR sites is not found, no entry clones will be produced.

When performing multi-site gateway cloning, *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 <https://www.thermofisher.com/dk/en/home/life-science/cloning/gateway-cloning/multisite-gateway-technology.html?SID=fr-gwcloning-3>

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 20.38.

You can choose to create a sequence list with the bi-products as well.

20.5 Gel electrophoresis

CLC Main Workbench enables the user to simulate the separation of nucleotide sequences on a gel. This feature is useful when 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 several ways to simulate gel separation of nucleotide sequences:

- When performing the **Restriction Site Analysis** from the Toolbox, you can choose to create

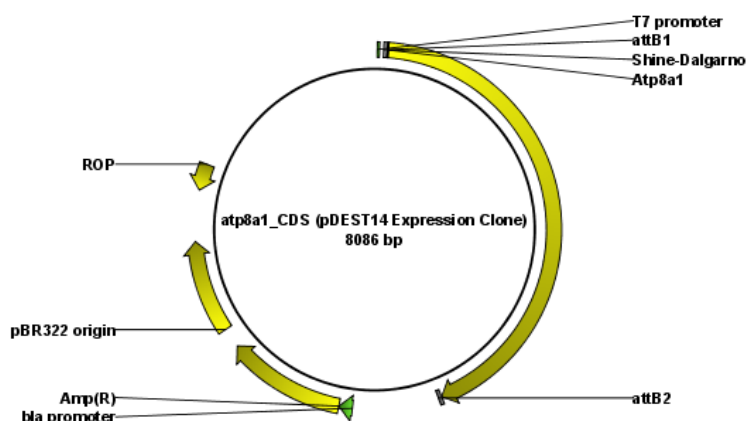


Figure 20.38: The resulting expression clone opened in a circular view.

a restriction map which can be shown as a gel (see section 20.1.2).

- From all the graphical views of sequences, you can right-click the name of the sequence and choose **Digest and Create Restriction Map** (☰). The sequence will be digested with the enzymes that are selected in the Side Panel. The views where this option is available are listed below:
 - Circular view (see section 11.2).
 - Ordinary sequence view (see section 11.1).
 - Graphical view of sequence lists (see section 11.6).
 - Cloning editor (see section 20.3).
 - Primer designer (see section 19.3).
- **Separate sequences on gel:** To separate sequences without restriction enzyme digestion, first create a sequence list of the sequences in question, then click the **Gel** button (☰) at the bottom of the view of the sequence list (figure 20.39).

20.5.1 Gel view

In figure 20.40 you can see a simulation of a gel with its Side Panel to the right.

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.

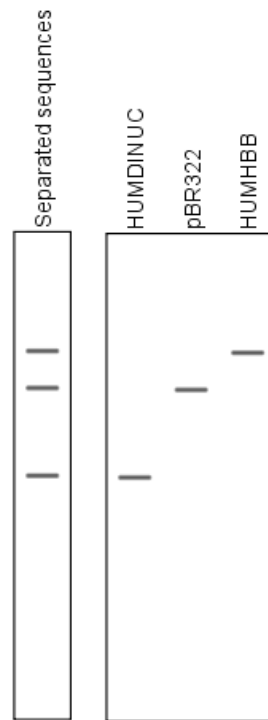


Figure 20.39: A sequence list shown as a gel.

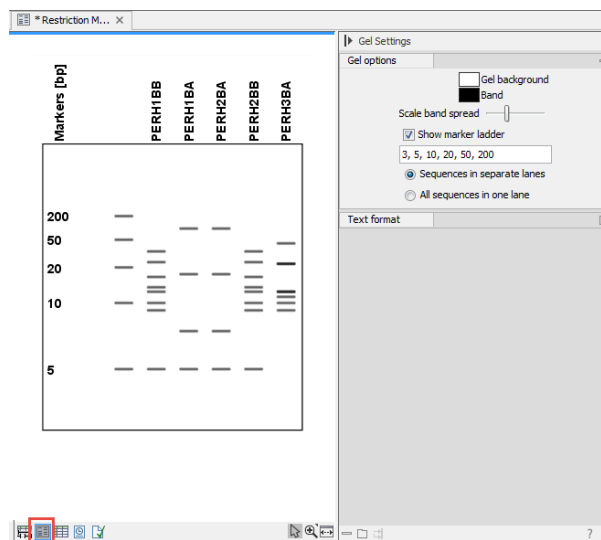


Figure 20.40: Five lanes showing fragments of five sequences cut with restriction enzymes.

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.

Chapter 21

RNA structure

Contents

21.1 RNA secondary structure prediction	431
21.1.1 Selecting sequences for prediction	431
21.1.2 Secondary structure prediction parameters	432
21.1.3 Structure as annotation	436
21.2 View and edit secondary structures	437
21.2.1 Graphical view and editing of secondary structure	437
21.2.2 Tabular view of structures and energy contributions	440
21.2.3 Symbolic representation in sequence view	442
21.2.4 Probability-based coloring	444
21.3 Evaluate structure hypothesis	445
21.3.1 Selecting sequences for evaluation	445
21.3.2 Probabilities	446
21.4 Structure scanning plot	446
21.4.1 Selecting sequences for scanning	447
21.4.2 The structure scanning result	448
21.5 Bioinformatics explained: RNA structure prediction by minimum free energy minimization	448
21.5.1 The algorithm	449
21.5.2 Structure elements and their energy contribution	451

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 known from crystallography but de novo prediction of tertiary structures is not possible with current methods. However, as for proteins RNA tertiary structures can be characterized by secondary structural elements which are hydrogen bonds within the molecule that form several recognizable "domains" of secondary structure like stems, hairpin loops, bulges and internal loops. A large part of the functional information is thus contained in the secondary structure of the RNA molecule, as shown by the high degree of base-pair conservation observed in the evolution of RNA molecules.

Computational prediction of RNA secondary structure is a well defined problem and a large body of work has been done to refine prediction algorithms and to experimentally estimate the relevant biological parameters.

In *CLC 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

21.1 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 section 21.5.

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 Mfold version 3, see <http://mfold.rna.albany.edu/?q=mfold/mfold-references>.

21.1.1 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 21.1.

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 sequence lists from the selected elements. You can use both DNA and RNA sequences - DNA will be folded as if it were RNA.

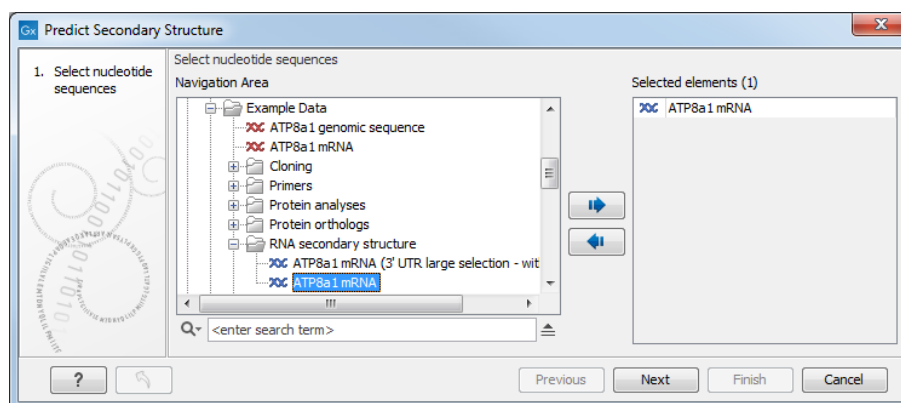


Figure 21.1: Selecting RNA or DNA sequences for structure prediction (DNA is folded as if it was RNA).

21.1.2 Secondary structure prediction parameters

Click **Next** to adjust secondary structure prediction parameters (figure 21.2).

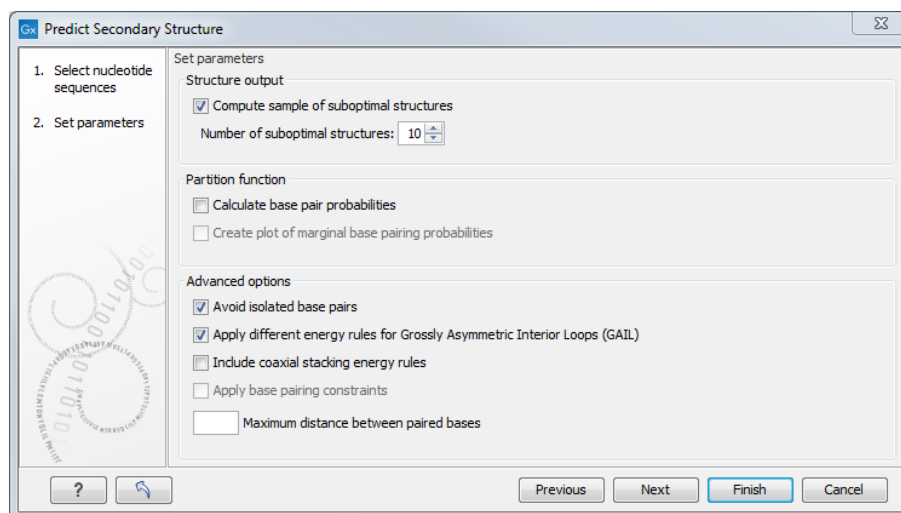


Figure 21.2: Adjusting parameters for secondary structure prediction.

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 **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].

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 21.3.

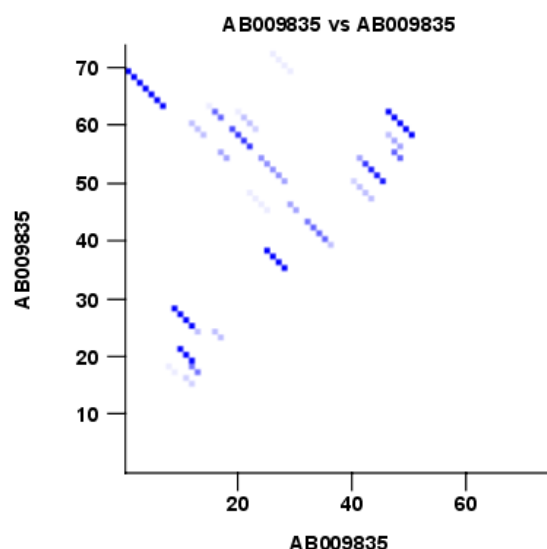


Figure 21.3: *The marginal base pair probability of all possible base pairs.*

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 21.2.3 and also in the secondary structure view. An example is shown in figure 21.4. Furthermore, the marginal probabilities are accessible from tooltips when hovering over the relevant parts of the structure.

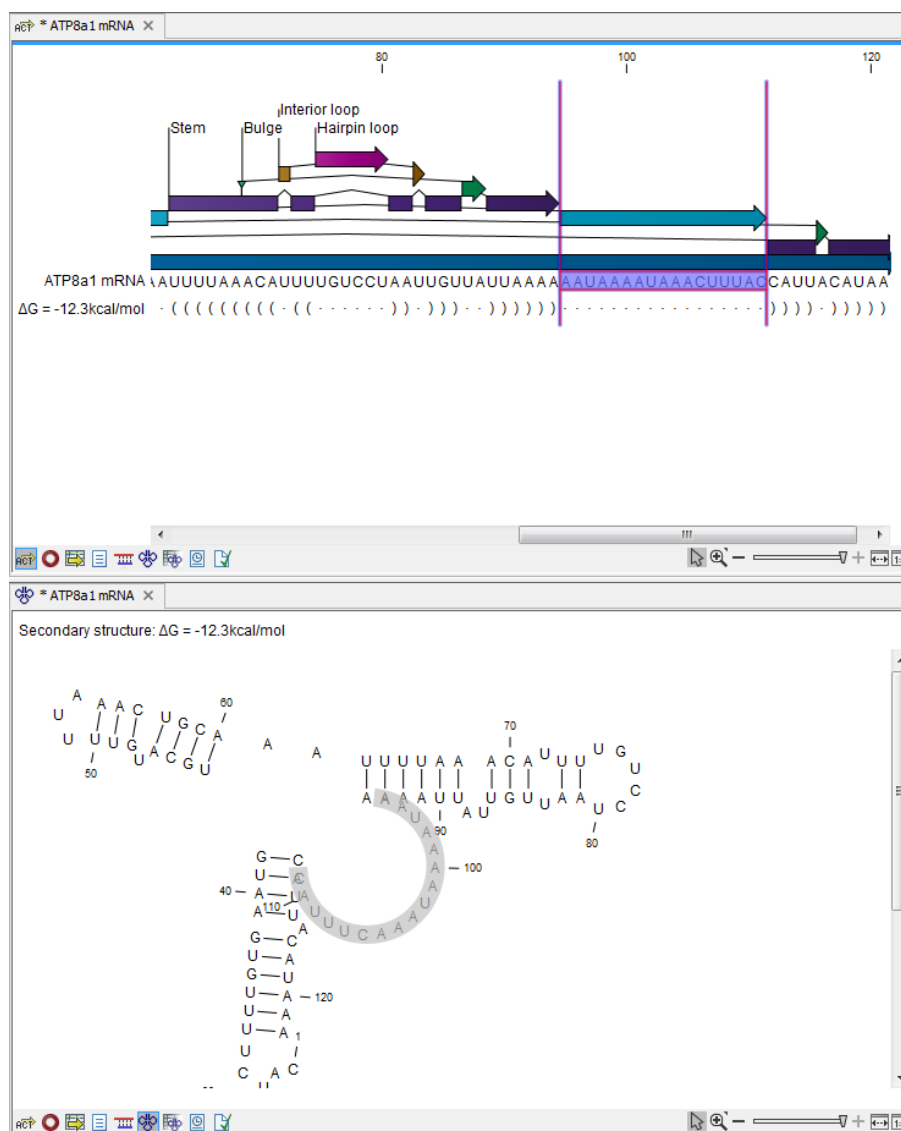


Figure 21.4: Marginal probability of base pairs shown in linear view (top) and marginal probability of being unpaired shown in the secondary structure 2D view (bottom).

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://mfold.rna.albany.edu/doc/mfold-manual/node5.php>).
- **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 21.1.2).

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 21.5).

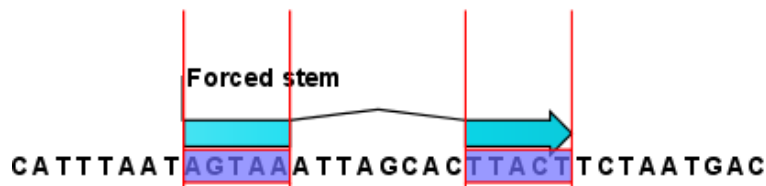


Figure 21.5: Force a stem of the selected bases.

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 21.6).

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:

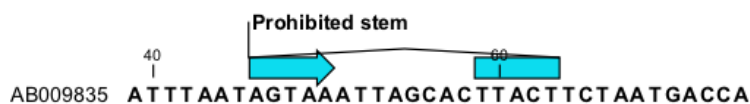


Figure 21.6: Prohibit the selected bases from forming a stem.

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 21.7.



Figure 21.7: 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.

21.1.3 Structure as annotation

You can choose to add the elements of the best structure as annotations (see figure 21.8).

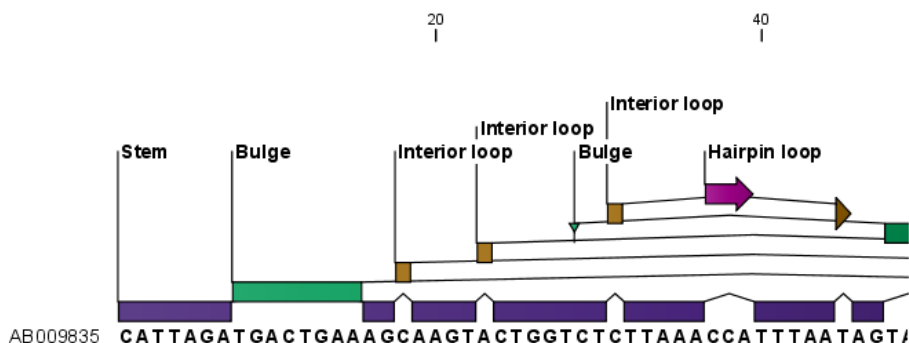


Figure 21.8: Annotations added for each structure element.

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 21.2.2.

21.2 View and edit secondary structures

When you predict RNA secondary structure (see section 21.1), the resulting predictions are attached to the sequence and can be shown as:

- Annotations in the ordinary sequence views (Linear sequence view (👉), Annotation table (📄) etc. This is only possible if this has been chosen in the dialog in figure 21.2. See an example in figure 21.8.
- Symbolic representation below the sequence (see section 21.2.3).
- A graphical view of the secondary structure (see section 21.2.1).
- 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 21.2.2.

21.2.1 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 21.9.

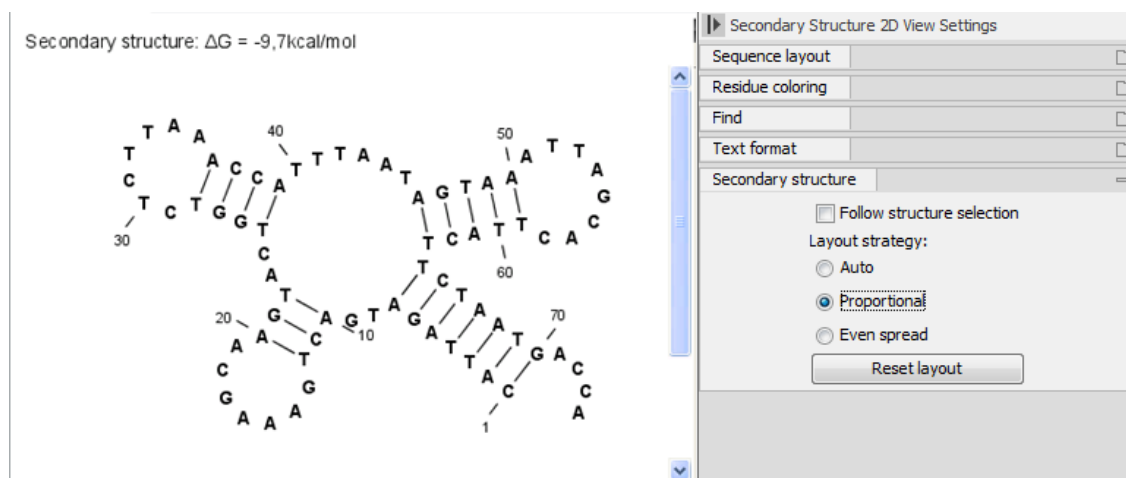


Figure 21.9: The secondary structure view of an RNA sequence zoomed in.

Like the normal sequence view, you can use **Zoom in** (🔍) and **Zoom out** (🔍). Zooming in will reveal the residues of the structure as shown in figure 21.9. 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 11.1.1. 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 21.2.2 for more information.
- **Layout strategy.** Specify the strategy used for the layout of the structure. In addition to these strategies, you can also modify the layout manually as explained in the next section.
 - **Auto.** The layout is adjusted to minimize overlapping structure elements [Han et al., 1999]. This is the default setting (see figure 21.10).
 - **Proportional.** Arc lengths are proportional to the number of residues (see figure 21.11). Nothing is done to prevent overlap.
 - **Even spread.** Stems are spread evenly around loops as shown in figure 21.12.
- **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.

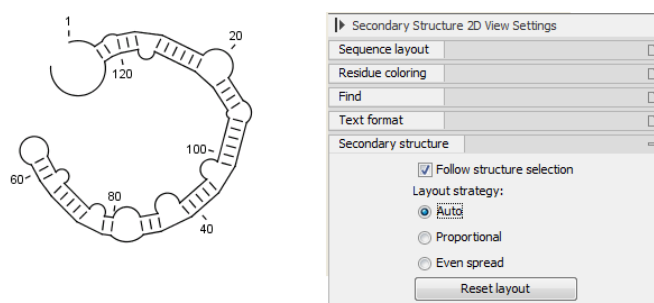


Figure 21.10: *Auto layout. Overlaps are minimized.*

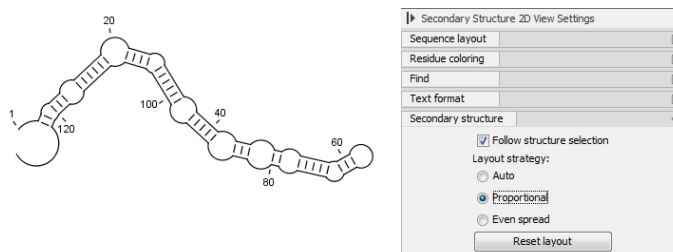


Figure 21.11: *Proportional layout. Length of the arc is proportional to the number of residues in the arc.*

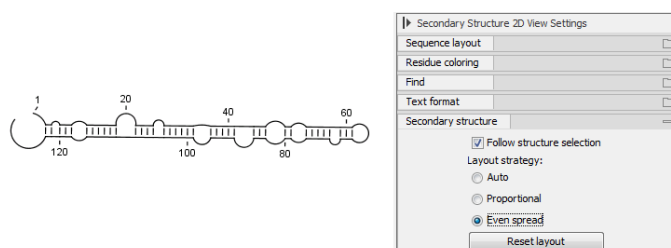


Figure 21.12: *Even spread. Stems are spread evenly around loops.*

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 21.13).

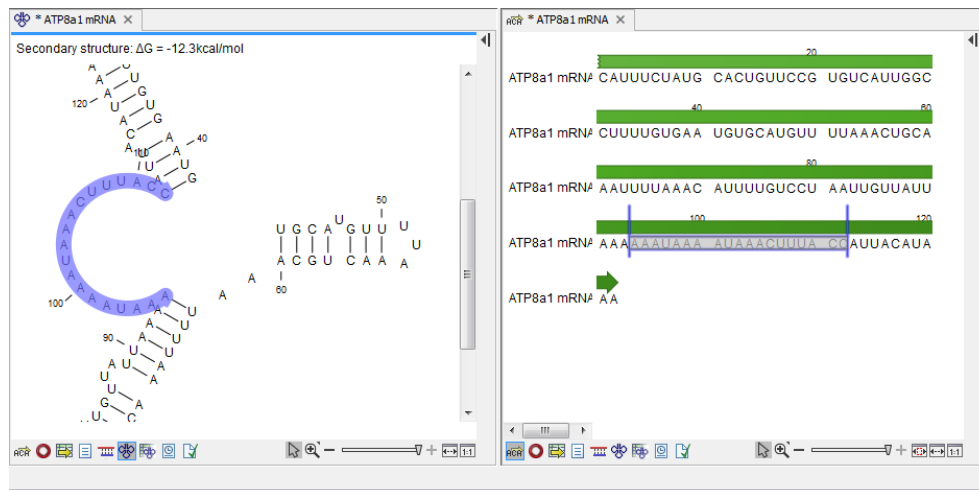


Figure 21.13: A split view of the secondary structure view and a linear sequence view.

If you make a selection in another sequence view, this will 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 (right-click on the zoom icon below the View Area). 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 21.14).

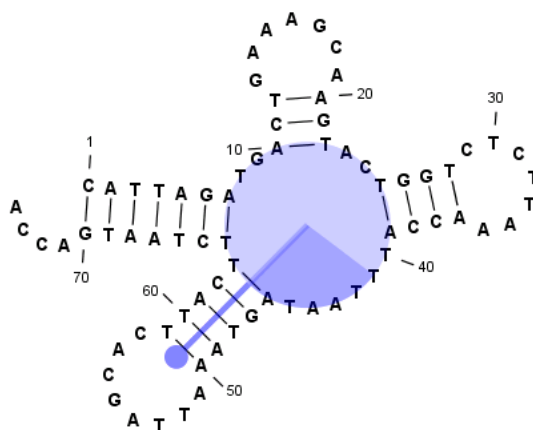


Figure 21.14: The blue circle represents the anchor point for rotating the substructure.

Click and drag to rotate the part of the structure represented by the line going from the anchor point. In order to keep the bases in a relatively sequential arrangement, there is a restriction

on how much the substructure can be rotated. The highlighted part of the circle represents the angle where rotating is allowed.

In figure 21.15, the structure shown in figure 21.14 has been modified by dragging with the mouse.

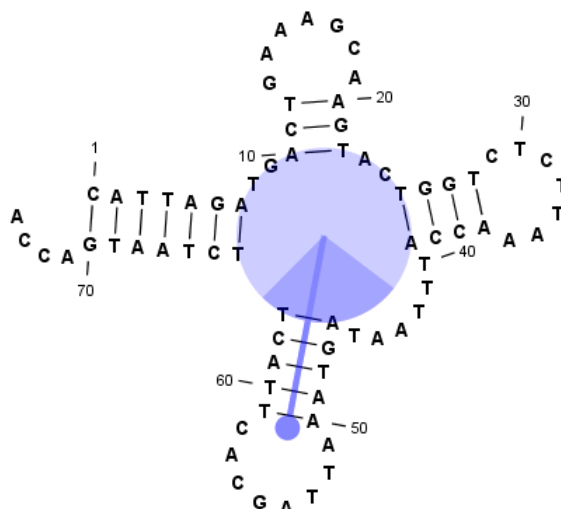


Figure 21.15: *The structure has now been rotated.*

Press **Reset layout** in the **Side Panel** to reset the layout to the way it looked when the structure was predicted.

21.2.2 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 21.1), 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 21.2.1).
- 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 21.16.

On the left side, all computed structures are listed with the information about structure name, when the structure was created, the free energy of the structure and the probability of the structure if the partition function was calculated. Selecting a row (equivalent: a structure) will display a tree of the contained substructures with their contributions to the total structure free energy. Each substructure contains a union of nested structure elements and other substructures (see a detailed description of the different structure elements in section 21.5.2). Each substructure

The screenshot shows a window titled 'ATP8a1 mRNA (... X)'. It contains a table with 11 rows and 4 columns: Name, Created, ΔG , and Probability. To the right of the table is a list of substructure elements for the selected structure, each with a small icon and a description.

Name	Created	ΔG	Probability
$\Delta G = -146,7\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.7kcal/mol	6.38E-17
$\Delta G = -146,7\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.7kcal/mol	6.38E-17
$\Delta G = -146,5\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.5kcal/mol	4.61E-17
$\Delta G = -146,5\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.5kcal/mol	4.61E-17
$\Delta G = -146,4\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.4kcal/mol	3.92E-17
$\Delta G = -146,4\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.4kcal/mol	3.92E-17
$\Delta G = -146,4\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.4kcal/mol	3.92E-17
$\Delta G = -146,3\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.3kcal/mol	3.33E-17
$\Delta G = -146,3\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.3kcal/mol	3.33E-17
$\Delta G = -146,3\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.3kcal/mol	3.33E-17
$\Delta G = -146,3\text{kcal/mol}$	23-Jun-2008 11:48:29	-146.3kcal/mol	3.33E-17

Elements of structure ' $\Delta G = -146,7\text{kcal/mol}$ ': $\Delta G = -146,7\text{kcal/mol}$

- Stem with bifurcation at 1..484: $\Delta G = -142,0\text{kcal/mol}$
- U-A helix end at (1, 484): $\Delta G = 0,5\text{kcal/mol}$
- Dangling A at 485, dangling from position 484: $\Delta G = -0,8\text{kcal/mol}$
- Dangling G at 487, dangling from position 488: $\Delta G = -0,2\text{kcal/mol}$
- Stem with hairpin at 488..505: $\Delta G = -2,8\text{kcal/mol}$
- U-A helix end at (488, 505): $\Delta G = 0,5\text{kcal/mol}$
- Dangling A at 506, dangling from position 505: $\Delta G = -0,8\text{kcal/mol}$
- Dangling A at 507, dangling from position 508: $\Delta G = -0,3\text{kcal/mol}$
- Stem with hairpin at 508..539: $\Delta G = -0,5\text{kcal/mol}$
- U-A helix end at (508, 539): $\Delta G = 0,5\text{kcal/mol}$
- Dangling A at 540, dangling from position 539: $\Delta G = -0,8\text{kcal/mol}$

Figure 21.16: The secondary structure table with the list of structures to the left, and to the right the substructures of the selected structure.

contributes a free energy given by the sum of its nested substructure energies and energies of its nested structure elements.

The substructure elements to the right are ordered after their occurrence in the sequence; they are described by a region (the sequence positions covered by this substructure) and an energy contribution. Three examples of mixed substructure elements are "Stem base pairs", "Stem with bifurcation" and "Stem with hairpin".

The "Stem base pairs"-substructure is simply a union of stacking elements. It is given by a joined set of base pair positions and an energy contribution displaying the sum of all stacking element-energies.

The "Stem with bifurcation"-substructure defines a substructure enclosed by a specified base pair with and with energy contribution ΔG . The substructure contains a "Stem base pairs"-substructure and a nested bifurcated substructure (multi loop). Also bulge and interior loops can occur separating stem regions.

The "Stem with hairpin"-substructure defines a substructure starting at a specified base pair with an enclosed substructure-energy given by ΔG . The substructure contains a "Stem base pairs"-substructure and a hairpin loop. Also bulge and interior loops can occur, separating stem regions.

In order to describe the tree ordering of different substructures, we use an example as a starting point (see figure 21.17).

The structure is a (disjoint) nested union of a "Stem with bifurcation"-substructure and a dangling nucleotide. The nested substructure energies add up to the total energy. The "Stem with bifurcation"-substructure is again a (disjoint) union of a "Stem base pairs"-substructure joining position 1-7 with 64-70 and a multi loop structure element opened at base pair(7,64). To see these structure elements, simply expand the "Stem with bifurcation" node (see figure 21.18).

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 21.18 the "Stem with bifurcation" is selected in the table, and this part of the structure is high-lighted in the Secondary Structure 2D view.

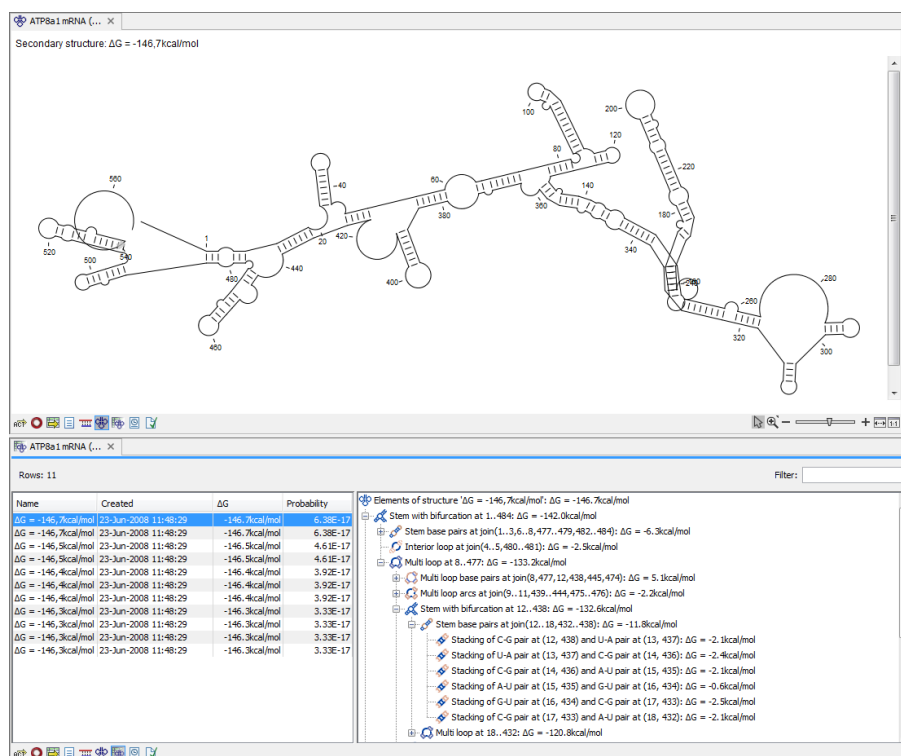


Figure 21.17: A split view showing a structure table to the right and the secondary structure 2D view to the left.

The correspondence between the table and the structure editor makes it easy to inspect the thermodynamic details of the structure while keeping a visual overview as shown in the above figures.

Handling multiple structures

The table to the left offers a number of tools for working with structures. Select a structure, right-click, and the following menu items will be available:

- **Open Secondary Structure in 2D View** (🔍). This will open the selected structure in the Secondary structure 2D view.
- **Annotate Sequence with Secondary Structure**. This will add the structure elements as annotations to the sequence. Note that existing structure annotations will be removed.
- **Rename Secondary Structure**. This will allow you to specify a name for the structure to be displayed in the table.
- **Delete Secondary Structure**. This will delete the selected structure.
- **Delete All Secondary Structures**. This will delete all the selected structures. Note that once you save and close the view, this operation is irreversible. As long as the view is open, you can **Undo** (↶) the operation.

21.2.3 Symbolic representation in sequence view

In the **Side Panel** of normal sequence views (ncp), you will find an extra group under **Nucleotide info** called **Secondary Structure**. This is used to display a symbolic representation of the

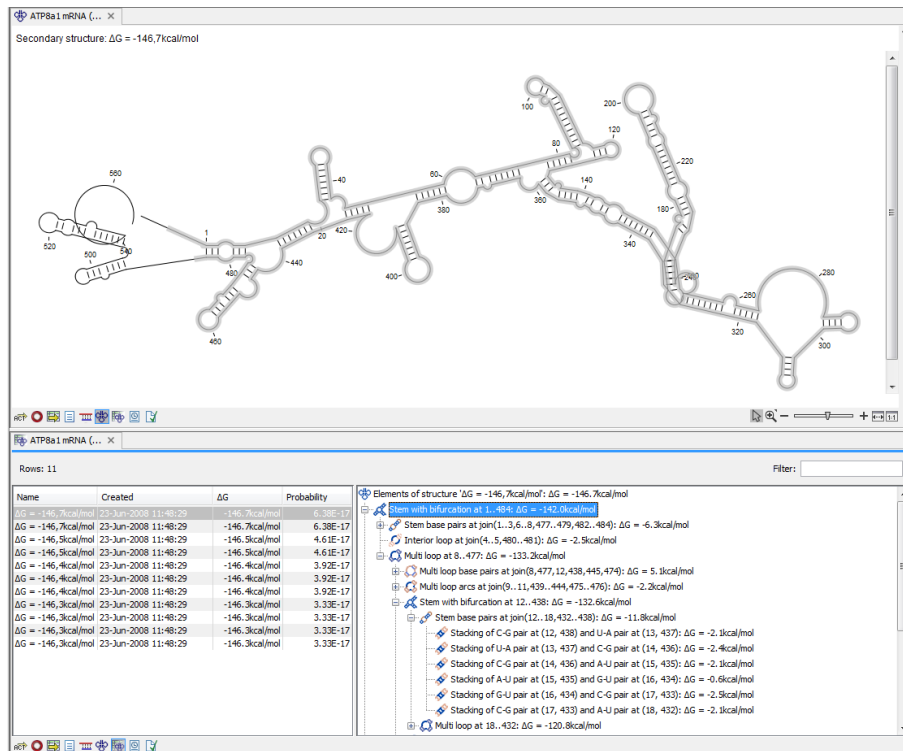


Figure 21.18: Now the "Stem with bifurcation" node has been selected in the table and a corresponding selection has been made in the view of the secondary structure to the left.

secondary structure along the sequence (see figure 21.19).

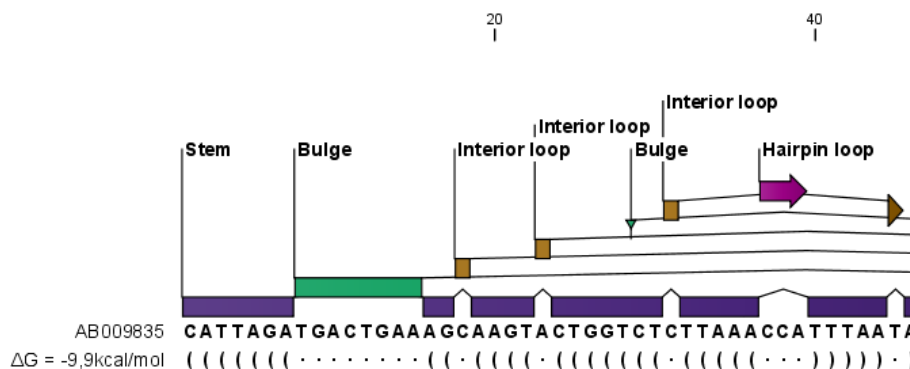


Figure 21.19: The secondary structure visualized below the sequence and with annotations shown above.

The following options can be set:

- **Show all structures.** If more than one structure is predicted, this option can be used if all the structures should be displayed.
- **Show first.** If not all structures are shown, this can be used to determine the number of structures to be shown.
- **Sort by.** When you select to display e.g. four out of eight structures, this option determines

which the "first four" should be.

- Sort by ΔG .
- Sort by name.
- Sort by time of creation.

If these three options do not provide enough control, you can rename the structures in a meaningful alphabetical way so that you can use the "name" to display the desired ones.

- **Base pair symbol.** How a base pair should be represented (see figure 21.19).
- **Unpaired symbol.** How bases which are not part of a base pair should be represented (see figure 21.19).
- **Height.** When you zoom out, this option determines the height of the symbols as shown in figure 21.20 (when zoomed in, there is no need for specifying the height).
- **Base pair probability.** See section 21.2.4 below).

When you zoom in and out, the appearance of the symbols change. In figure 21.19, the view is zoomed in. In figure 21.20 you see the same sequence zoomed out to fit the width of the sequence.

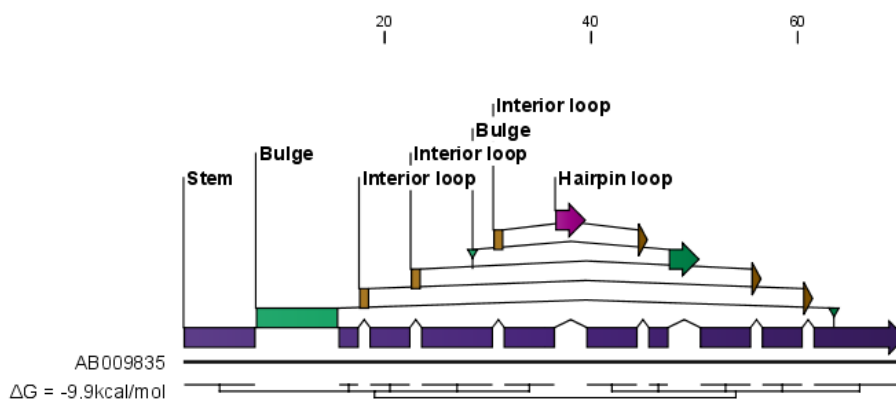


Figure 21.20: The secondary structure visualized below the sequence and with annotations shown above. The view is zoomed out to fit the width of the sequence.

21.2.4 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 21.4.

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.

21.3 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 21.1.2. By adding several annotations complex structural hypotheses can be formulated (see 21.21).

Given the set S of all possible structures, only a subset of these S_H will comply with the formulated hypotheses. We can now find the probability of H as:

$$P(H) = \frac{\sum_{s_H \in S_H} P(s_H)}{\sum_{s \in S} P(s)} = \frac{PF_H}{PF_{\text{full}}},$$

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. 21.21.

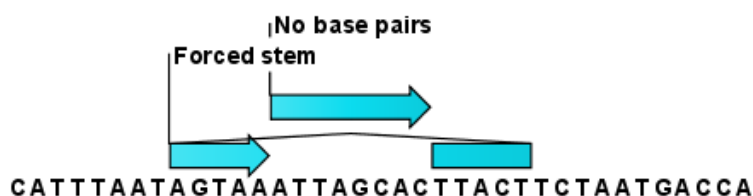


Figure 21.21: Two constraints defining a structural hypothesis.

21.3.1 Selecting sequences for evaluation

The evaluation is started from the **Toolbox**:

Toolbox | RNA Structure (🔍) | Evaluate Structure Hypothesis (🔍?)

This opens the dialog shown in figure 21.22.

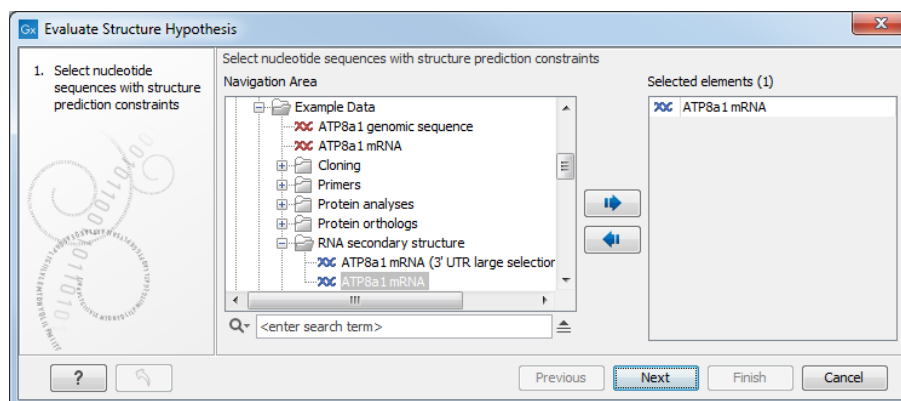


Figure 21.22: Selecting RNA or DNA sequences for evaluating structure hypothesis.

If you have selected sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements. Note, that the selected sequences must contain a structure hypothesis in the form of manually added constraint annotations.

Click **Next** to adjust evaluation parameters (see figure 21.23).

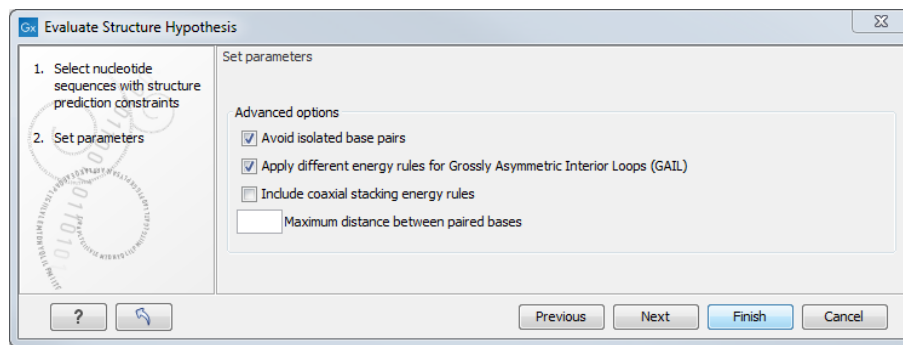


Figure 21.23: Adjusting parameters for hypothesis evaluation.

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://mfold.rna.albany.edu/doc/mfold-manual/node5.php>)
- **Include coaxial stacking energy rules.** Include free energy increments of coaxial stacking for adjacent helices [Mathews et al., 2004].

21.3.2 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 21.24).

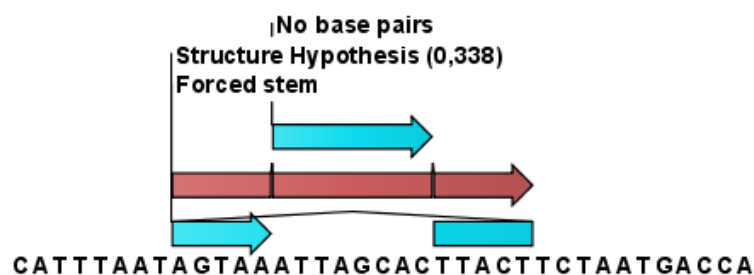


Figure 21.24: This hypothesis has a probability of 0.338 as shown in the annotation.

21.4 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 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].

21.4.1 Selecting sequences for scanning

The scanning is started from the **Toolbox**:

Toolbox | RNA Structure (📄) | Evaluate Structure Hypothesis (📈)

This opens the dialog shown in figure 21.25.

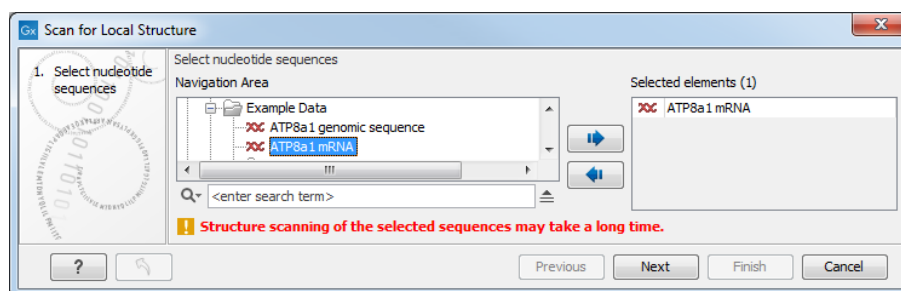


Figure 21.25: Selecting RNA or DNA sequences for structure scanning.

If you have selected sequences before choosing the Toolbox action, they are now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove sequences or sequence lists from the selected elements.

Click **Next** to adjust scanning parameters (see figure 21.26).

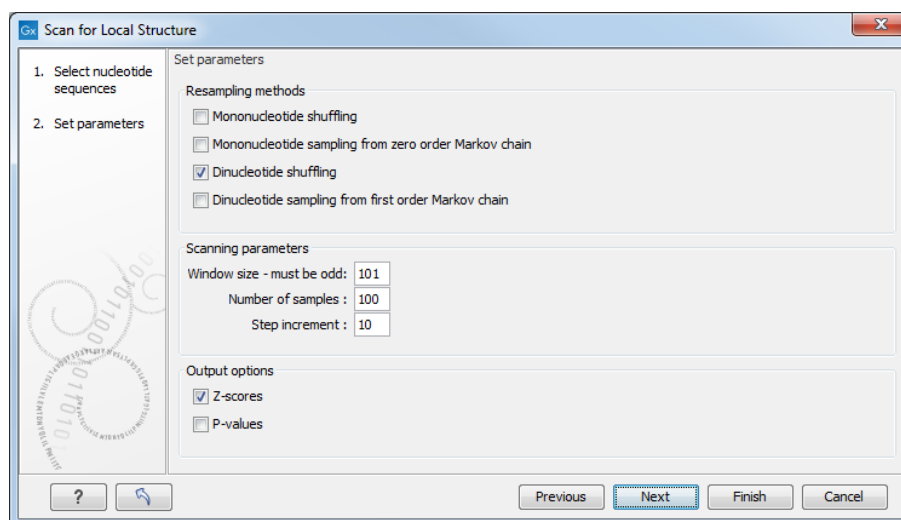


Figure 21.26: Adjusting parameters for structure scanning.

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.

21.4.2 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 21.27).

21.5 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.

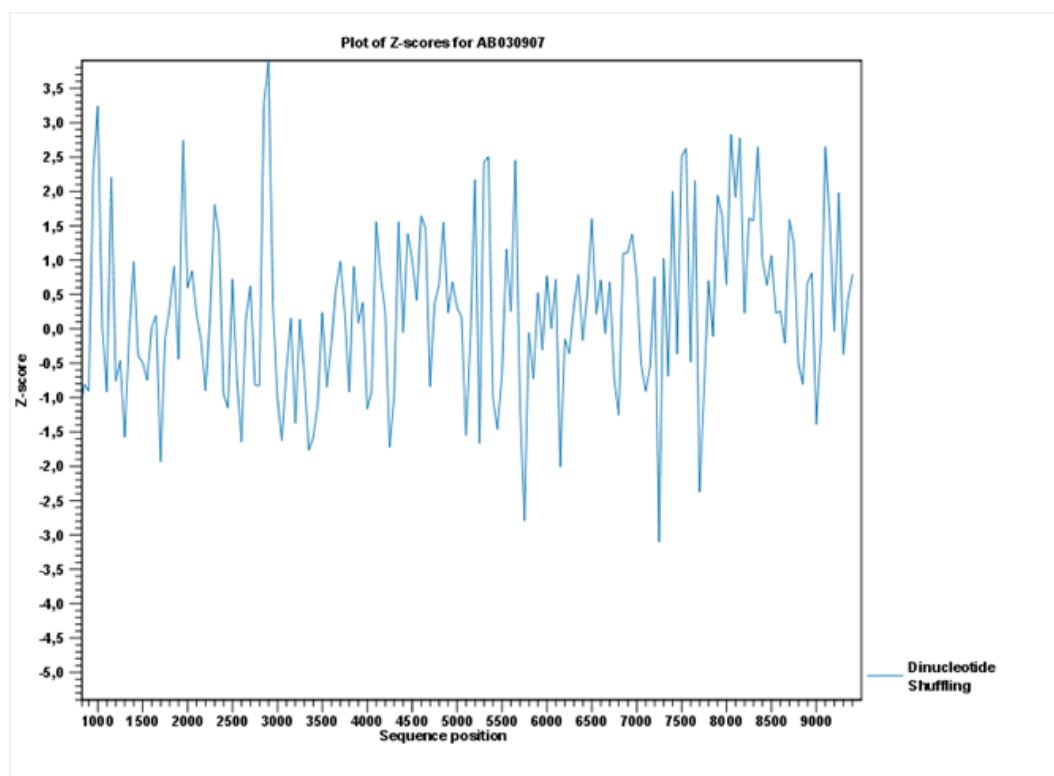


Figure 21.27: A plot of the Z-scores produced by sliding a window along a sequence.

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].

21.5.1 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 21.28.

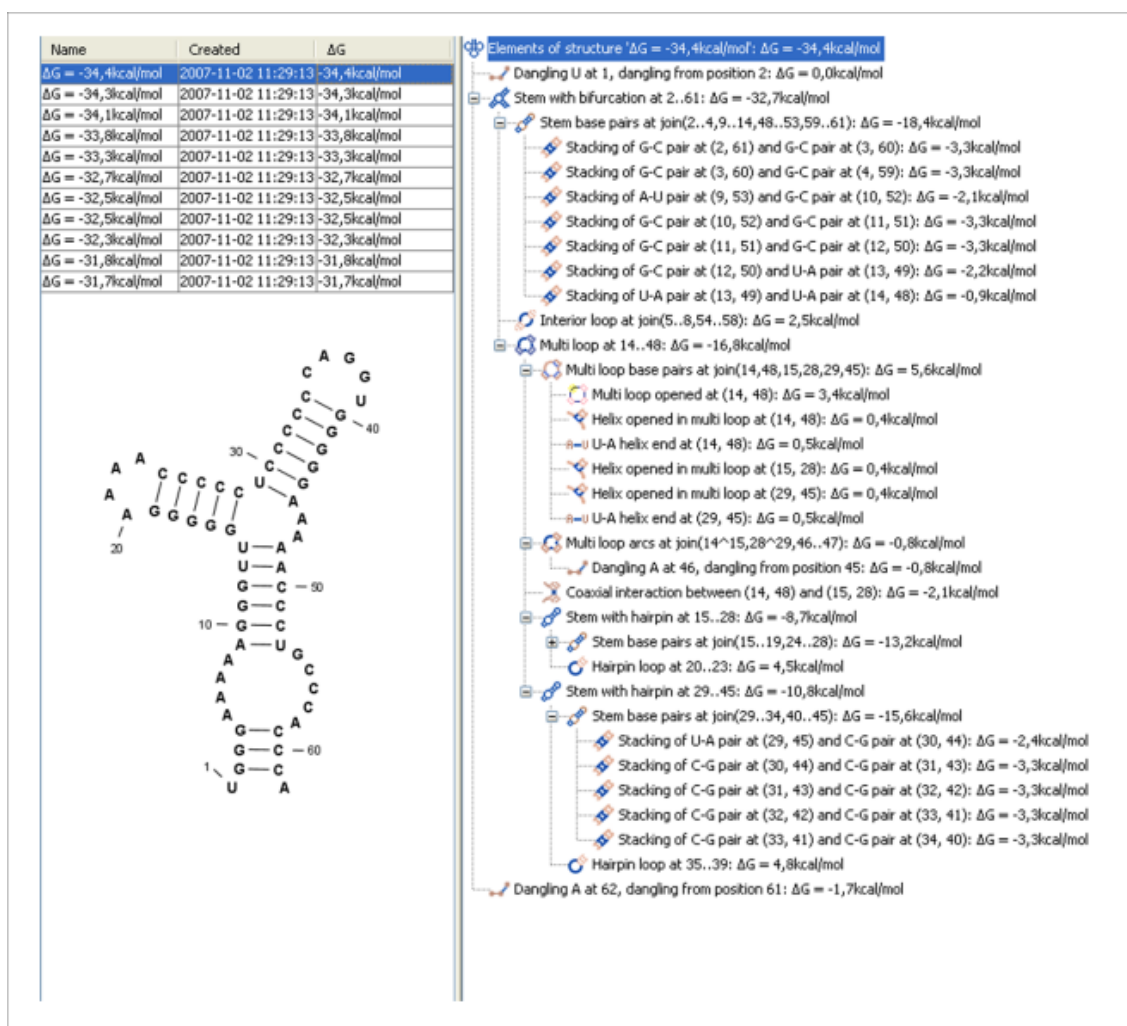


Figure 21.28: A number of suboptimal structures have been predicted using **CLC Main Workbench** and are listed at the top left. At the right hand side, the structural components of the selected structure are listed in a hierarchical structure and on the left hand side the structure is displayed.

21.5.2 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 base pairs, we can define the following distinct structure

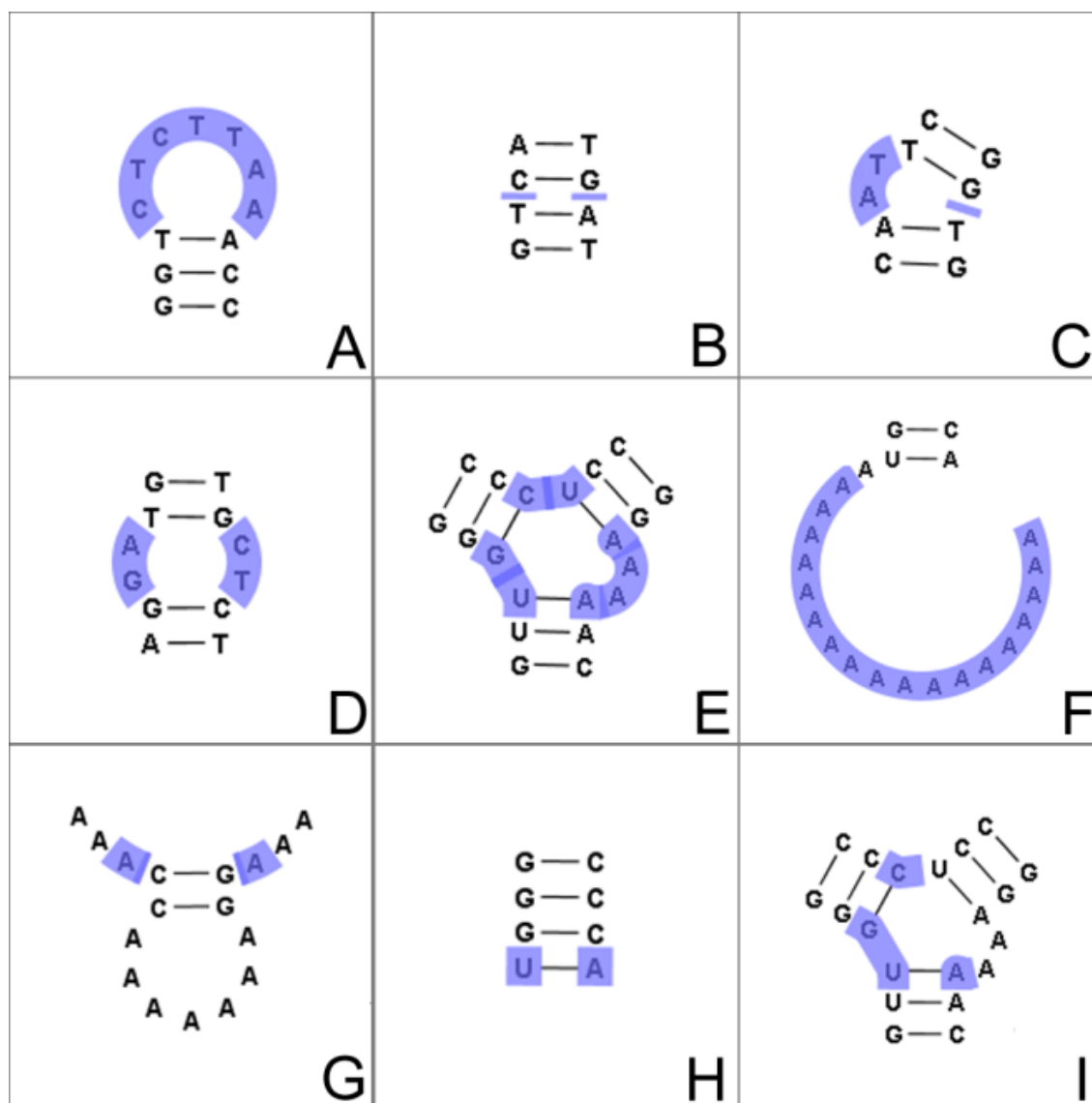





Figure 21.29: The different structure elements of RNA secondary structures predicted with the free energy minimization algorithm in **CLC Main Workbench**. See text for a detailed description.




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 21.29A).
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 21.29B). 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 21.29C). 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 21.29D). 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 21.29E). 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 21.29F). 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 21.29G). The energy contribution is determined by the single stranded nucleotide, its orientation and on the adjacent base pair.
- **Non-GC terminating stem** (). 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 21.29H).
- **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 21.29I). 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 21.30) and it has been shown that this can dramatically increase the fidelity of the secondary structure prediction [Mathews and Turner, 2006].



Figure 21.30: *Known structural features can be added as constraints to the secondary structure prediction algorithm in **CLC Main Workbench**.*

Chapter 22

Expression analysis

Contents

22.1 Experimental design	456
22.1.1 Setting up an experiment	456
22.1.2 Organization of the experiment table	459
22.1.3 Adding annotations to an experiment	465
22.1.4 Scatter plot view of an experiment	466
22.1.5 Cross-view selections	468
22.2 Transformation and normalization	469
22.2.1 Selecting transformed and normalized values for analysis	469
22.2.2 Transformation	470
22.2.3 Normalization	471
22.3 Quality control	473
22.3.1 Creating box plots - analyzing distributions	473
22.3.2 Hierarchical clustering of samples	476
22.3.3 Principal component analysis	481
22.4 Statistical analysis - identifying differential expression	486
22.4.1 Empirical analysis of DGE	486
22.4.2 Tests on proportions	491
22.4.3 Gaussian-based tests	492
22.4.4 Corrected p-values	494
22.4.5 Volcano plots - inspecting the result of the statistical analysis	495
22.5 Feature clustering	497
22.5.1 Hierarchical clustering of features	498
22.5.2 K-means/medoids clustering	502
22.6 Annotation tests	505
22.6.1 Hypergeometric tests on annotations	505
22.6.2 Gene set enrichment analysis	508
22.7 General plots	511
22.7.1 Histogram	511
22.7.2 MA plot	513
22.7.3 Scatter plot	517

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.

22.1 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 in section I.

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.

22.1.1 Setting up an experiment

To set up an experiment:

Toolbox | Expression Analysis (📁) | Set Up Experiment (🛠️)

Select the samples that you wish to use by double-clicking or selecting and pressing the **Add** (➡️) button (see figure 22.1).

Clicking **Next** shows the dialog in figure 22.2.

Here you define the experiment type and the number of groups in the experiment.

The options are:

- **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.

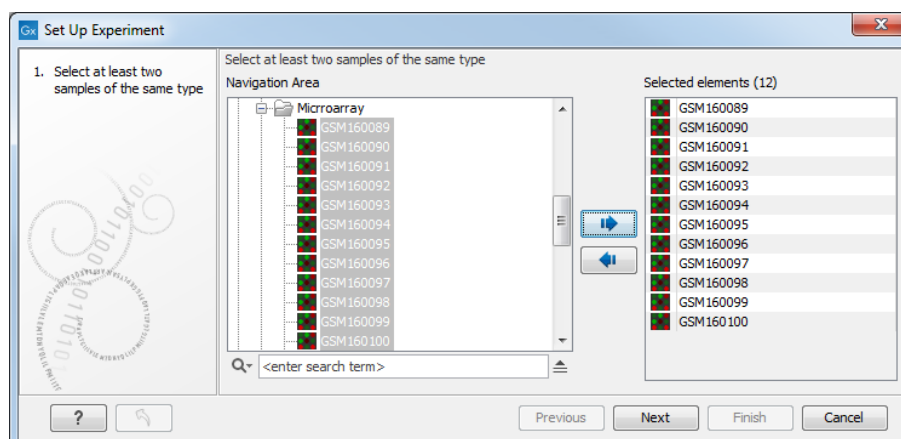


Figure 22.1: Select the samples to use for setting up the experiment.

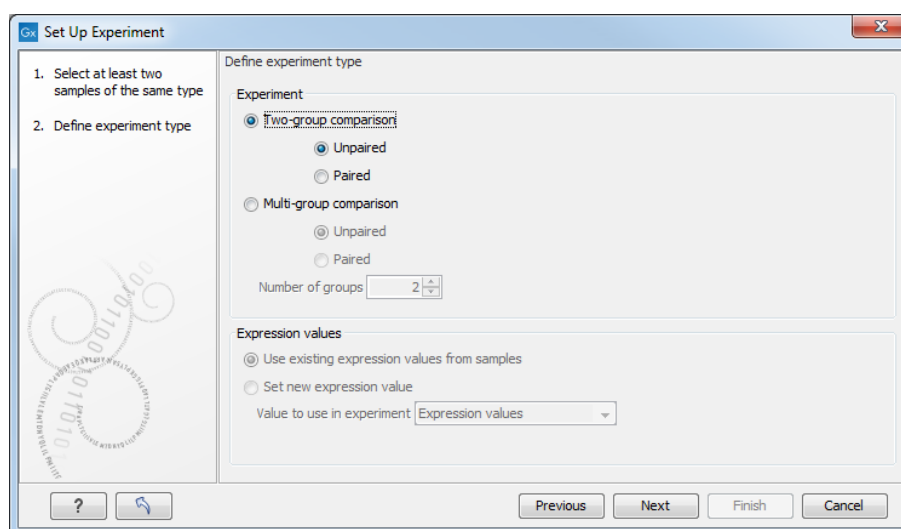


Figure 22.2: Defining the number of groups and expression value type.

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 **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.

- **Expression values.** If you choose to **Set new expression value** you can choose between the following options depending on whether you look at the gene or transcript level:
 - **Genes: Unique exon reads.** The number of reads that match uniquely to the exons (including the exon-exon and exon-intron junctions).
 - **Genes: Unique gene reads.** This is the number of reads that match uniquely to the gene.
 - **Genes: Total exon reads.** Number of reads mapped to this gene that fall entirely within an exon or in exon-exon or exon-intron junctions. As for the "Total gene reads"

this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon of this gene.

- **Genes: Total gene reads.** This is all the reads that are mapped to this gene, i.e., both reads that map uniquely to the gene and reads that matched to more positions in the reference (but fewer than the "Maximum number of hits for a read" parameter) which were assigned to this gene.
- **Genes: RPKM.** This is the expression value measured in RPKM [Mortazavi et al., 2008]: $RPKM = \frac{\text{total exon reads}}{\text{mapped reads (millions)} \times \text{exon length (KB)}}$. See exact definition below. Even if you have chosen the RPKM values to be used in the **Expression values** column, they will also be stored in a separate column. This is useful to store the RPKM if you switch the expression measure.
- **Transcripts: Unique transcript reads.** This is the number of reads in the mapping for the gene that are uniquely assignable to the transcript. This number is calculated after the reads have been mapped and both single and multi-hit reads from the read mapping may be unique transcript reads.
- **Transcripts: Total transcript reads.** Once the "Unique transcript read's" have been identified and their counts calculated for each transcript, the remaining (non-unique) transcript reads are assigned randomly to one of the transcripts to which they match. The "Total transcript reads" counts are the total number of reads that are assigned to the transcript once this random assignment has been done. As for the random assignment of reads among genes, the random assignment of reads within a gene but among transcripts, is done proportionally to the "unique transcript counts" normalized by transcript length, that is, using the RPKM. Unique transcript counts of 0 are not replaced by 1 for this proportional assignment of non-unique reads among transcripts.
- **Transcripts: RPKM.** The RPKM value for the transcript, that is, the number of reads assigned to the transcript divided by the transcript length and normalized by "Mapped reads" (see below).

Clicking **Next** shows the dialog in figure 22.3.

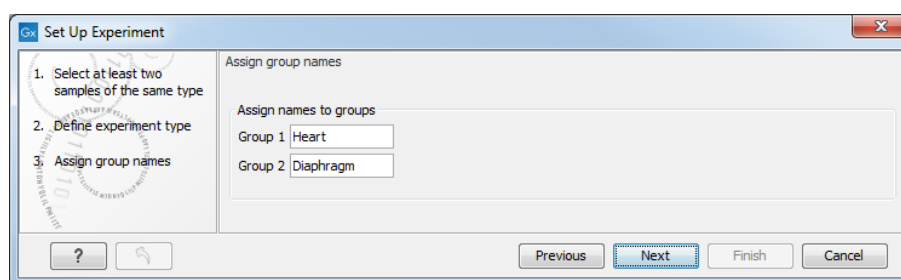


Figure 22.3: Naming the groups.

Depending on the number of groups selected in figure 22.2, 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** (X) button. If you need more groups, simply click **Add New Group**.

Click **Next** when you have named the groups, and you will see figure 22.4.

This is where you define which group the individual sample belongs to. Simply select one or

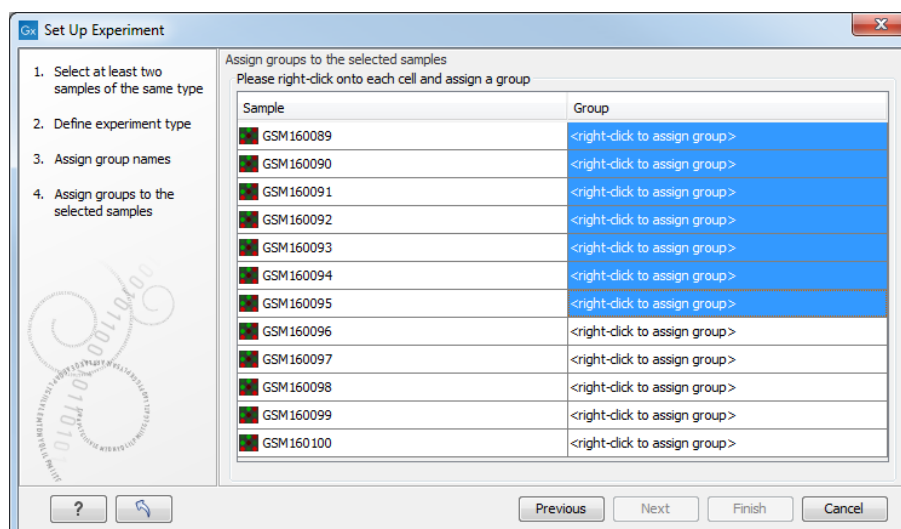


Figure 22.4: Putting the samples into groups.

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 22.2, 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 **Finish** to start the tool.

22.1.2 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 included is described in the sections below.

When you open it, it is shown in the experiment table (see figure 22.5).

Feature ID	Experiment					GSM160089		GSM160090	
	Total presen...	Range (origi...	IQR (original...	Difference (...	Fold Change...	Expression values	Presence call	Expression values	Presence
1367452_at	12	862.00	470.50	464.82	1.19	2,532.90	P	2,518.60	P
1367453_at	12	1,231.00	430.80	428.37	1.13	3,464.20	P	3,197.40	P
1367454_at	12	536.50	349.10	153.85	1.10	1,620.80	P	1,870.50	P
1367455_at	12	2,196.20	1,352.90	-772.35	-1.17	5,512.50	P	4,103.90	P
1367456_at	12	2,095.50	1,264.20	-1,205.65	-1.27	6,090.80	P	5,352.20	P
1367457_at	12	508.20	319.00	-64.73	-1.07	1,093.90	P	1,134.30	P
1367458_at	12	268.30	148.90	112.30	1.38	347.80	P	223.90	P
1367459_at	12	3,993.80	2,434.30	2,557.35	1.36	7,665.80	P	7,415.90	P
1367460_at	12	1,182.80	557.00	-84.73	-1.17	3,155.70	P	2,946.90	P
1367461_at	12	485.70	280.20	184.35	1.29	507.00	P	610.30	P
1367462_at	12	1,032.50	309.70	268.23	1.08	3,207.50	P	3,371.30	P
1367463_at	12	1,621.60	510.00	701.97	1.21	3,510.30	P	3,050.30	P
1367464_at	12	317.70	202.00	-111.67	-1.13	797.70	P	1,038.90	P
1367465_at	12	699.00	196.80	257.50	1.22	1,103.10	P	1,281.80	P
1367466_at	12	265.50	122.50	-54.67	-1.04	1,385.10	P	1,321.30	P

Figure 22.5: Opening the experiment.

For a general introduction to table features like sorting and filtering, see section 3.3.

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 4.6).

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 22.6).

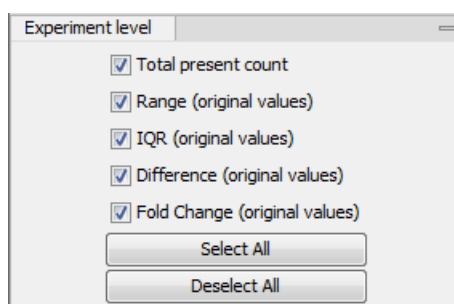


Figure 22.6: The initial view of the experiment level for a two-group experiment.

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. Entries of plus or minus infinity in the 'Fold Change' columns of the Experiment area represent those where one of the expression values in the calculation is a 0. 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 [22.2.3](#) and [22.2.2](#), 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 [22.4.5](#). 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

The results of each statistical test performed are in the columns listed in this area. In the table, a heading is given for each test. Information about the results of statistical tests are described in the statistical analysis section (see section 22.4).

An example of Analysis level settings is shown in figure 22.7.

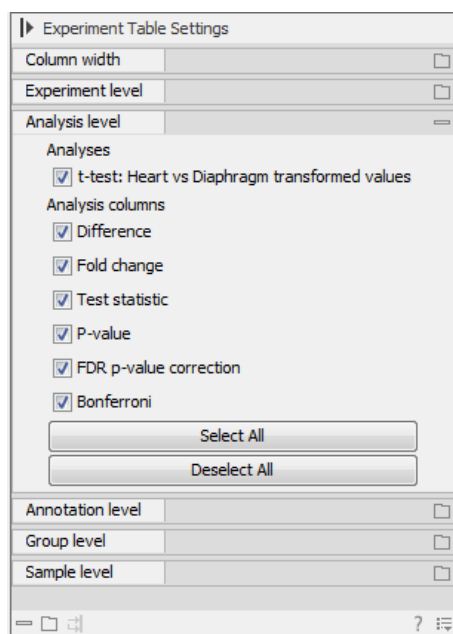


Figure 22.7: An example of columns available under the Analysis level section.

Note: Some column names here are the same as ones under the Experiment level, but the results here are from statistical tests, while those under the Experiment level section are calculations carried out directly on the expression levels.

Annotation level

If your experiment is annotated (see section 22.1.3), the annotations will be listed in the **Annotation level** group as shown in figure 22.8.

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 4.6).

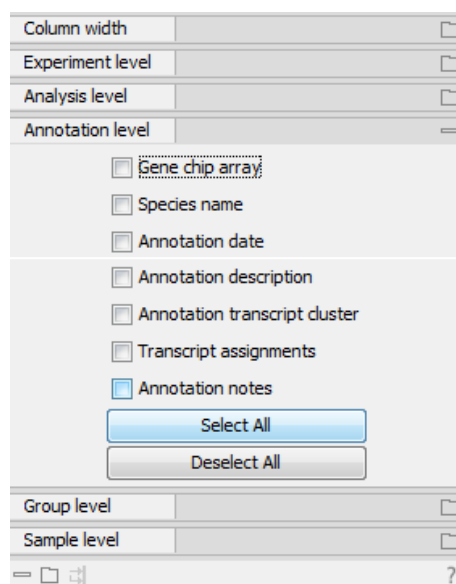


Figure 22.8: An annotated experiment.

Group level

At the group level, you can show/hide entire groups (*Heart* and *Diaphragm* in figure 22.5). 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 22.2.3 and 22.2.2, respectively), the means of the normalized and transformed values will also appear.

An example is shown in figure 22.9.

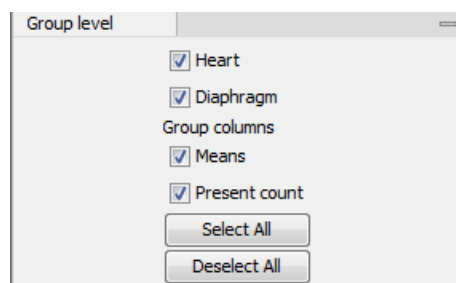


Figure 22.9: Group level .

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 22.2.3 and 22.2.2, respectively), the normalized and transformed values will also appear.

An example is shown in figure 22.10.

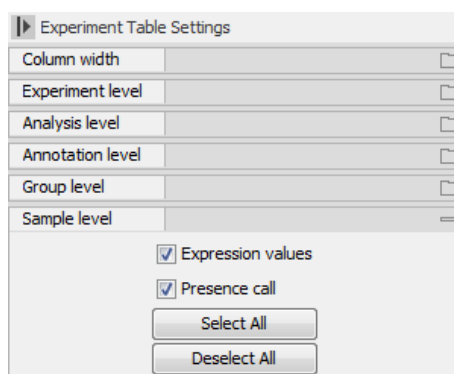


Figure 22.10: Sample level when transformation and normalization has been performed.

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 3.3).

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 22.11).

1	1160	186	341	175	330
1	1212	100	794	85	767
1	795	506	559	498	549
1	1116	427	438	421	422
1	3732	965	970	930	934
1	1827	68	68	64	64
1	2391	1840	1874	1816	1846
1	1635	28	35	14	14
1	6292	715	740	626	630
1	667	262	262	262	262

Figure 22.11: 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 22.12).

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 11. You can now use the downloaded sequences for further analysis in the Workbench.

1	1160	186	341	175	330
1	1212	100	794	85	767
1	795	506	559	498	549
1	1116	427	438	421	422
1	3732	965	970	930	934
1	1827	68	68	64	64
1	2391	1840	1874	1816	1846
1	1635	28	35	14	14
1	6292	715	740	626	630
1	667	267	262	267	262

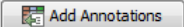
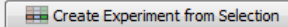









Figure 22.12: Select sequences and press the download button.

22.1.3 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 I 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 I. In order to associate an annotation file with an experiment, either select the annotation file when you set up the experiment (see section 22.1.1), or click:

Toolbox | Expression Analysis  | **Annotation Test** | **Add Annotations** 

Select the experiment  and the annotation file  and click **Finish**. You will now be able to see the annotations in the experiment as described in section 22.1.2. You can also add annotations by pressing the **Add Annotations**  button at the bottom of the table (see figure 22.13).

1	1160	186	341	175	330
1	1212	100	794	85	767
1	795	506	559	498	549
1	1116	427	438	421	422
1	3732	965	970	930	934
1	1827	68	68	64	64
1	2391	1840	1874	1816	1846
1	1635	28	35	14	14
1	6292	715	740	626	630
1	667	267	262	267	262

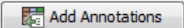
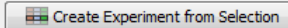





Figure 22.13: Adding annotations by clicking the button at the bottom of the experiment table.

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 22.14).

Add Annotations

1. Select experiments and one annotation table

2. Set parameters

Set parameters

Choose columns to match

Experiment or sample column to match on: Feature ID

Annotation column to match on: Feature ID

Remove leading zeros

?
Previous
Next
Finish
Cancel

Figure 22.14: Choosing how to match annotations with samples.

In this dialog, you can specify how to match the annotations to the features in the sample. The Workbench looks at the columns in the annotation file and lets you choose which column that should be used for matching to the feature IDs in the experimental data (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.

22.1.4 Scatter plot view of an experiment

At the bottom of the experiment table, you can switch between different views of the experiment (see figure 22.15).

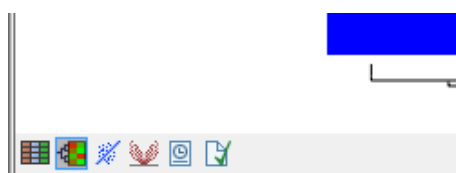


Figure 22.15: An experiment can be viewed in several ways.

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 22.16.

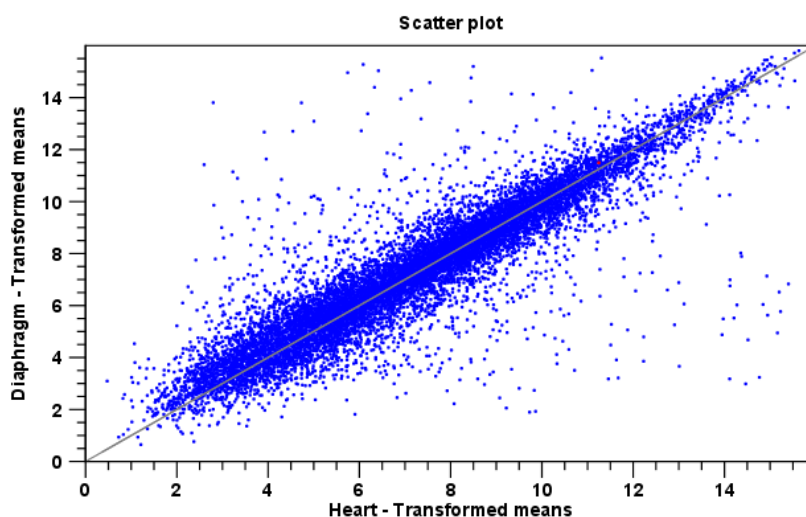


Figure 22.16: A scatter plot of group means for two groups (transformed expression values).

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.
- **Show Pearson correlation** When checked, the Pearson correlation coefficient (r) is displayed on the plot.

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 - **Values to plot** - is where you choose the values to be displayed in the graph. The default for a two-group experiment is to plot the group means.

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 4.6).

22.1.5 Cross-view selections

There are a number of different ways of looking at an experiment as shown in figure 22.17).



Figure 22.17: 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 22.17, 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 22.18.

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 heatmap, 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 22.1.2) you typically want to choose 'Difference'.

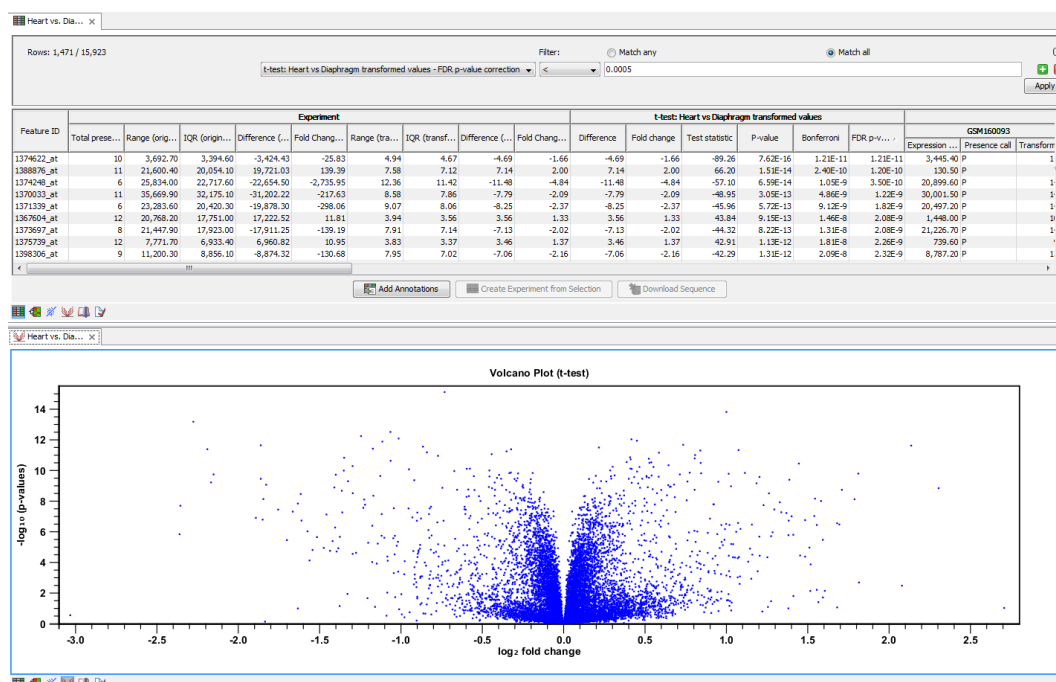


Figure 22.18: A split view showing an experiment table at the top and a volcano plot at the bottom (note that you need to perform statistical analysis to show a volcano plot, see section 22.4).

22.2 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).

22.2.1 Selecting transformed and normalized values for analysis

A number of the tools for Expression Analysis use the following expression level values: *Original*, *Transformed* and *Normalized* (figure 22.19).

In this case, the values have not been normalized, so it is not possible to select normalized values.

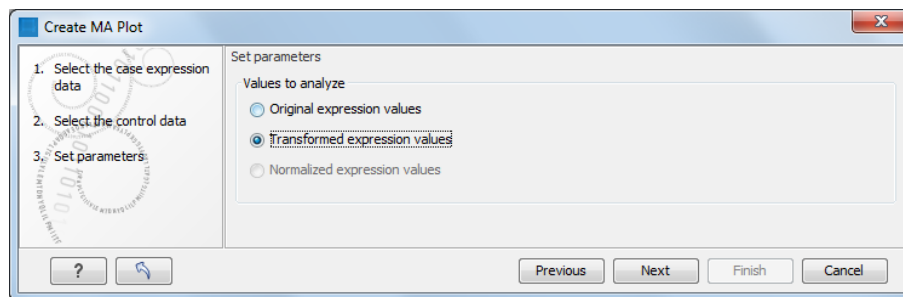


Figure 22.19: Selecting which version of the expression values to analyze. In this case, the values have not been normalized, so it is not possible to select normalized values.

22.2.2 Transformation

The *CLC Main Workbench* lets you transform expression values based on logarithm and adding a constant:

Toolbox | Expression Analysis (📊) | Transformation and Normalization | Transform (🔧)

Select a number of samples (📊) or (🇺🇸) or an experiment (📋) and click **Next**.

This will display a dialog as shown in figure 22.20.

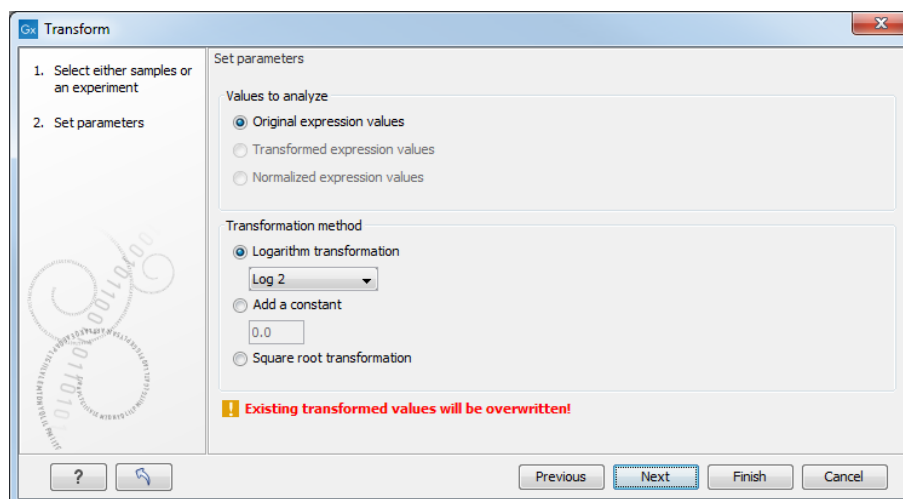


Figure 22.20: Transforming expression values.

At the top, you can select which values to transform (see section 22.2.1).

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 **Finish** to start the tool.

22.2.3 Normalization

The *CLC Main Workbench* lets you normalize expression values.

To start the normalization:

Toolbox | Expression Analysis (📁) | Transformation and Normalization | Normalize (🔧)

Select a number of samples (📁) or (📁) or an experiment (📁) and click **Next**.

This will display a dialog as shown in figure 22.21.

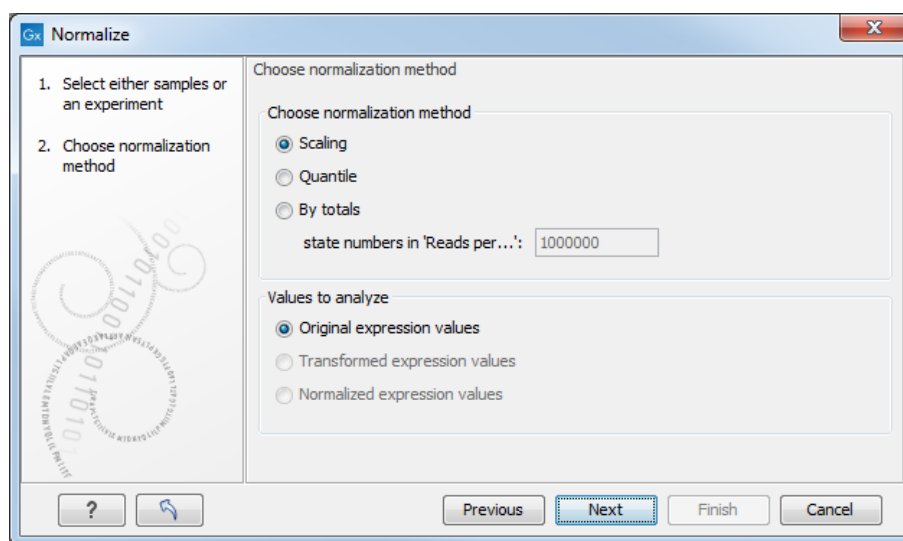


Figure 22.21: Choosing normalization method.

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 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 22.22 and 22.23 show the effect on the distribution of expression values when using scaling or quantile normalization, respectively.

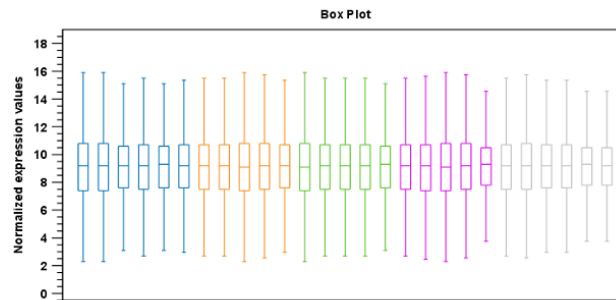


Figure 22.22: Box plot after scaling normalization.

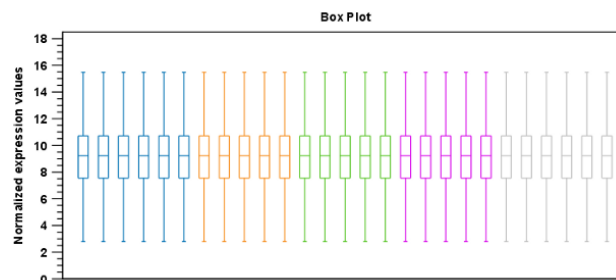


Figure 22.23: Box plot after quantile normalization.

At the bottom of the dialog in figure 22.21, you can select which values to normalize (see section 22.2.1).

Clicking **Next** will display a dialog as shown in figure 22.24.

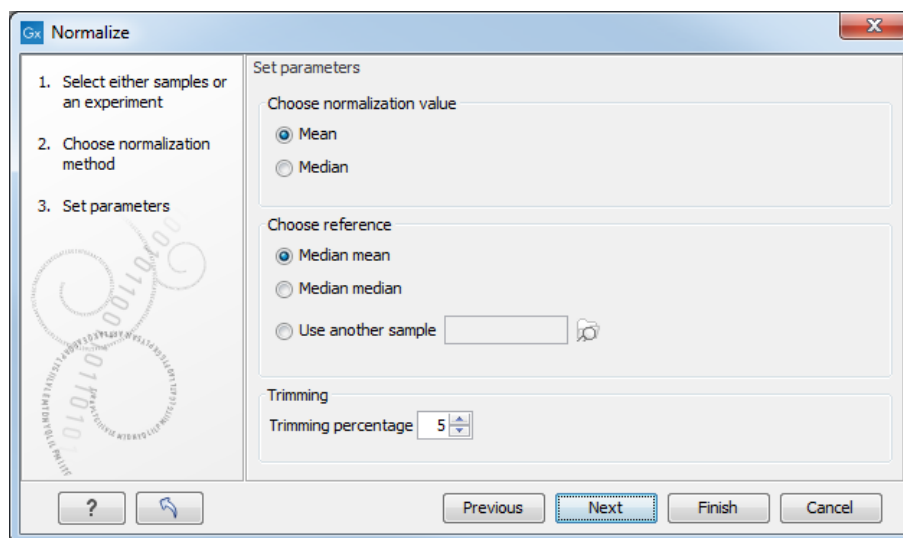


Figure 22.24: Normalization settings.

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 **Finish** to start the tool.

22.3 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.

22.3.1 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 its values is presented by a line representing a center, a box representing the middle part, and whiskers representing the tails of the distribution. Differences in the overall distributions of the samples in a study may indicate that normalization is required before the samples are comparable. An atypical distribution for a single sample (or a few samples), relative to the remaining samples in a study, could be due to imperfections in the preparation and processing of the sample, and may lead you to reconsider using the sample(s).

To create a box plot:

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

Select a number of samples () or () or an experiment () and click **Next**.

This will display a dialog as shown in figure [22.25](#).

Here you select which values to use in the box plot (see section [22.2.1](#)).

Click **Finish** to start the tool.

Viewing box plots

An example of a box plot of a two-group experiment with 12 samples is shown in figure [22.26](#).

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 [22.26](#)).

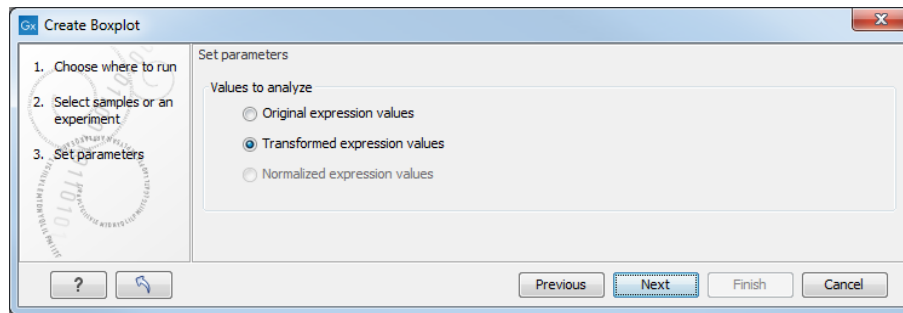


Figure 22.25: Choosing values to analyze for the box plot.

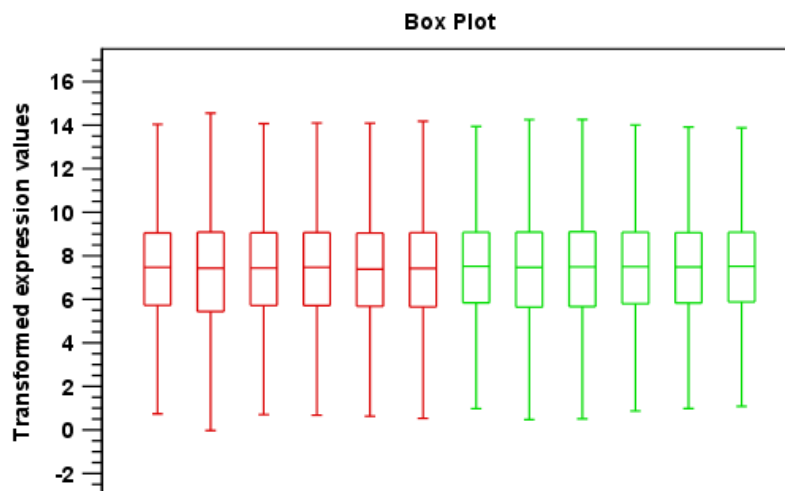


Figure 22.26: A box plot of 12 samples in a two-group experiment, colored by group.

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 22.27).

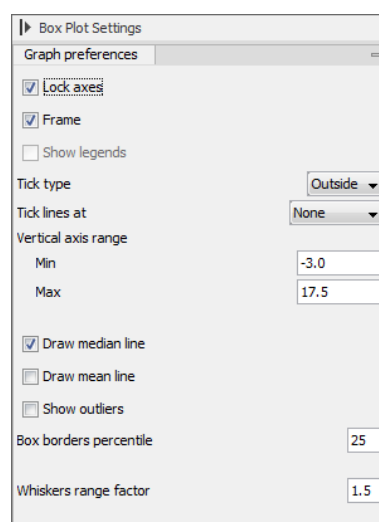


Figure 22.27: Graph preferences for a box plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
 - Outside
 - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
 - None
 - Major ticks
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Draw median line.** This is the default - the median is drawn as a line in the box.
- **Draw mean line.** Alternatively, you can also display the mean value as a line.
- **Show outliers.** The values outside the whiskers range are called outliers. Per default they are not shown. Note that the dot type that can be set below only takes effect when outliers are shown. When you select and deselect the **Show outliers**, the vertical axis range is automatically re-calculated to accommodate the new values.

Below the general preferences, you find the **Lines and dots** preferences, where you can adjust coloring and appearance (see figure 22.28).

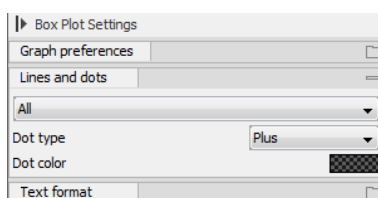


Figure 22.28: Lines and dot preferences for a box plot.

- **Select sample or group.** When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.
- **Dot type**
 - None

- Cross
 - Plus
 - Square
 - Diamond
 - Circle
 - Triangle
 - Reverse triangle
 - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a box plot, you need to save the settings of the **Side Panel** (see section 4.6).

Interpreting the box plot

This section will show how to interpret a box plot through a few examples.

First, if you look at figure 22.29, you can see a box plot for an experiment with 5 groups and 27 samples.

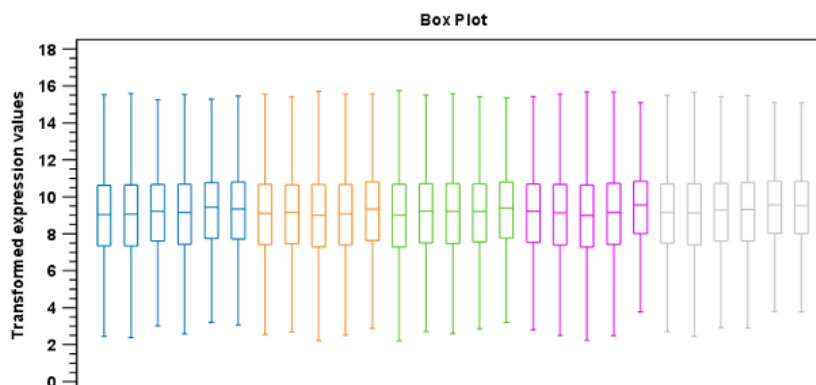


Figure 22.29: Box plot for an experiment with 5 groups and 27 samples.

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 22.30 shows a box plot for the same experiment after quantile normalization: the distributions have been brought into par.

In figure 22.31 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.

22.3.2 Hierarchical clustering of samples

A hierarchical clustering of samples is a tree representation of their relative similarity.

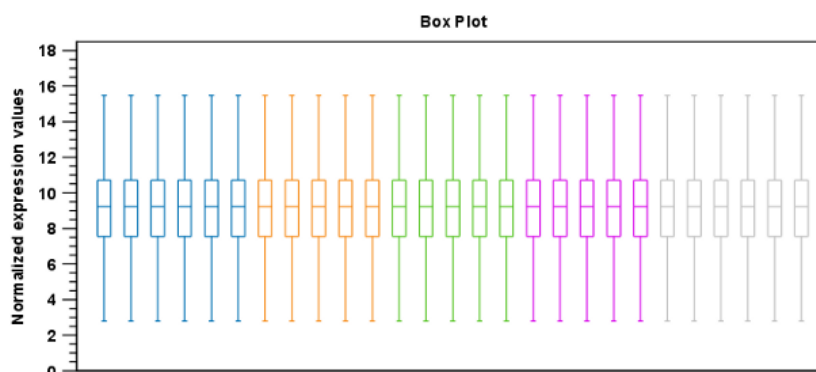


Figure 22.30: Box plot after quantile normalization.

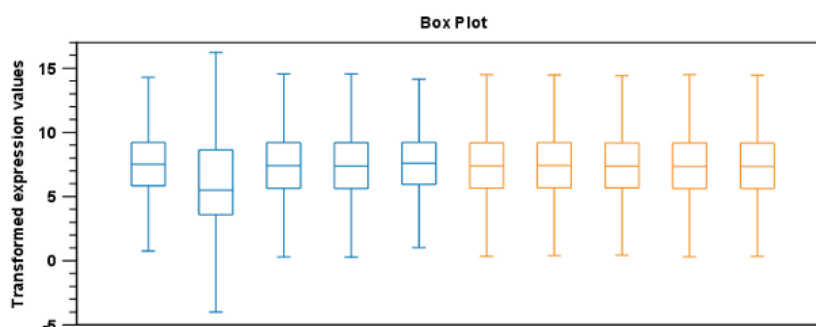


Figure 22.31: Box plot for a two-group experiment with 5 samples.

The tree structure is generated by

1. letting each sample 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 | Expression Analysis (📄) | Quality Control | Hierarchical Clustering of Samples (📊)

Select a number of samples (📄) or (📄) or an experiment (📄) and click **Next**.

This will display a dialog as shown in figure 22.32. 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.

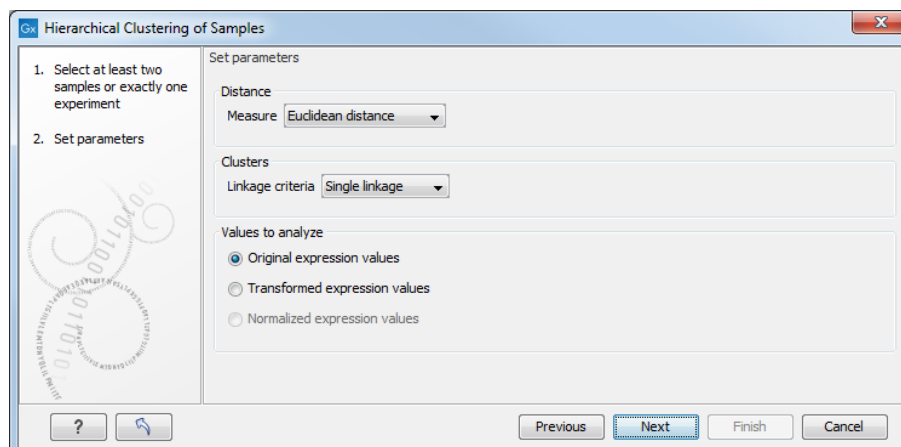


Figure 22.32: Parameters for hierarchical clustering of samples.

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 $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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 $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$ is defined as

$$r = \frac{1}{n-1} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s_x} \right) * \left(\frac{y_i - \bar{y}}{s_y} \right)$$

where \bar{x}/\bar{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 $1 - |\text{Pearson correlation}|$ 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 $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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 22.2.1).

Click **Finish** to start the tool.

Note: To be run on a server, the tool has to be included in a workflow, and the results will be displayed in a stand-alone new heat map rather than added into the input experiment table.

Result of hierarchical clustering of samples

The result of a sample clustering is shown in figure 22.33.

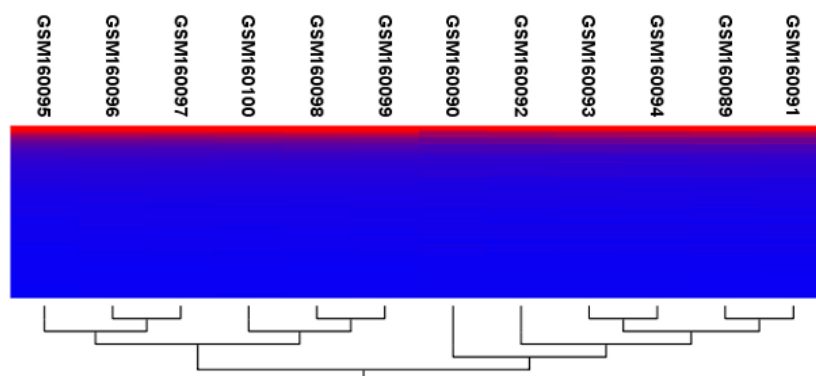


Figure 22.33: Sample clustering.

If you have used an **experiment** (📊) and ran the non-workflow version of the tool, 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 22.34).



Figure 22.34: Showing the hierarchical clustering of an experiment.

If you have run the workflow version of the tool, or 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 22.33, 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 22.5.1).

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, researchers 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 unclear 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 22.35).

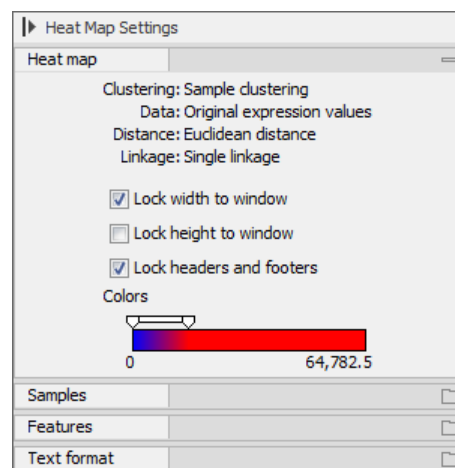


Figure 22.35: Side Panel of heat map.

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 22.48).

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 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.

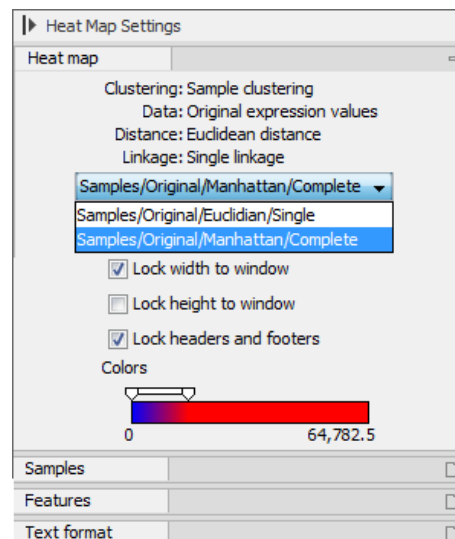


Figure 22.36: When more than one clustering has been performed, there will be a list of heat maps to choose from.

- **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 4.6).

22.3.3 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 either by finding the eigenvectors and eigenvalues of the *covariance matrix* of the samples or the *correlation matrix* of the samples (the correlation matrix is a 'normalized' version of the covariance matrix: the entries in the covariance matrix look like this $Cov(X, Y)$, and those in the correlation matrix like this: $Cov(X, Y)/(sd(X) * sd(Y))$). A covariance maybe any value, but a correlation is always between -1 and 1).

The eigenvectors are orthogonal. The first principal component is the eigenvector with the largest eigenvalue, and specifies the direction with the largest variability in the data. The second principal component is the eigenvector with the second largest eigenvalue, and specifies the direction with the second largest variability. Similarly for the third, etc. The data can be projected onto the space spanned by the eigenvectors. A plot of the data in the space spanned by the first and

second principal component will show a simplified version of the data with variability in other directions than the two major directions of variability ignored.

To start the analysis:

Toolbox | Expression Analysis () | Quality Control | Principal Component Analysis ()

Select a number of samples () or an experiment () and click **Next**.

This will display a dialog as shown in figure 22.37.

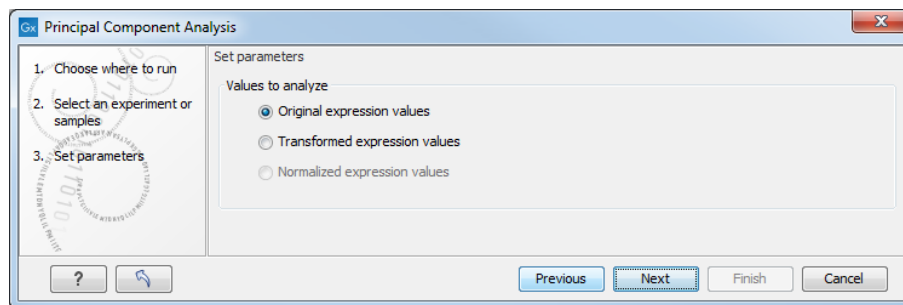


Figure 22.37: Selecting which values the principal component analysis should be based on.

In this dialog, you select the values to be used for the principal component analysis (see section 22.2.1).

Click **Finish** to start the tool.

Principal component analysis plot

This will create a principal component plot as shown in figure 22.38.

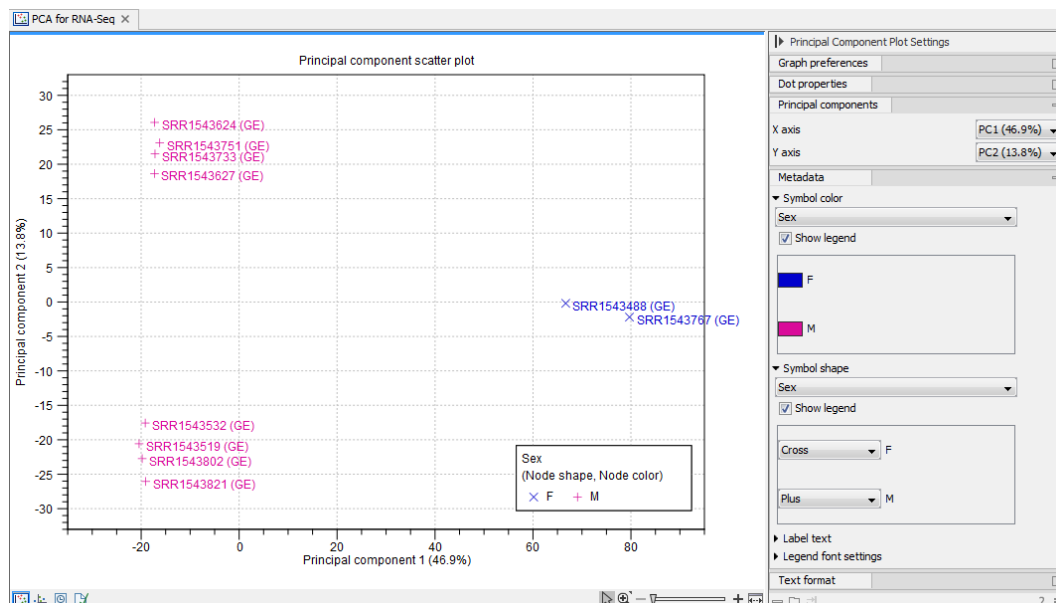


Figure 22.38: A principal component analysis colored by group.

The plot shows the projection of the samples onto the two-dimensional space spanned by the first

and second principal component of the covariance matrix. In the bottom part of the side-panel, the 'Projection/Correlation' part, you can change to show the projection onto the *correlation* matrix rather than the *covariance* matrix by choosing 'Correlation scatter plot'. Both plots will show how the samples separate along the two directions between which the samples exhibit the largest amount of variation. For the 'projection scatter plot' this variation is measured in absolute terms, and depends on the units in which you have measured your samples. The correlation scatter plot is a normalized version of the projection scatter plot, which makes it possible to compare principal component analysis between experiments, even when these have not been done using the same units (e.g an experiment that uses 'original' scale data and another one that uses 'log-scale' data).

The plot in figure 22.38 is based on a two-group experiment. The group relationships are indicated by color. We expect the samples from within a group to exhibit less variability when compared, than samples from different groups. Thus samples should cluster according to groups and this is what we see. The PCA plot is thus helpful in identifying outlying samples and samples that have been wrongly assigned to a group.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
 - Outside
 - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
 - None
 - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **y = 0 axis**. Draws a line where $y = 0$. Below there are some options to control the appearance of the line:
 - **Line width**
 - * Thin
 - * Medium
 - * Wide

- **Line type**

- * None
- * Line
- * Long dash
- * Short dash


- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties**:

- **Drop down menu** In this you choose which of the samples (that is, which 'dots') the choices you make below should apply to. You can choose between 'All', a particular group in your experiment, or a particular samples in your experiment.
- **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 4.6).

Scree plot

Besides the view shown in figure 22.38, 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 each of the principal components. The first principal component accounts for the largest part 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 4.6).

22.4 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 22.4.1 and section 22.4.2) and tests that assume that the data is real-valued, has Gaussian distributions and compare means (described in section 22.4.3).

To run the statistical analysis:

Expression Analysis () | **Statistical Analysis** | **Empirical Analysis of DGE** ()

Expression Analysis () | **Statistical Analysis** | **Proportion-based Statistical Analysis** ()

or **Expression Analysis** () | **Statistical Analysis** | **Gaussian Statistical Analysis** ()

For all kinds of statistical analyses, you first select the experiment () that you wish to use and click **Next** (learn more about setting up experiments in section 22.1.1).

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.

22.4.1 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.

The test uses the raw counts, and implicitly carries out normalization and transformation of these counts (see below for details). It 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 test is also appropriate for larger numbers of samples.

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 set of count samples against those in another set 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 be 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 22.4.1).

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` `tol` parameter, where the default is more stringent than that of edgeR. The advantage of using a more stringent value for this parameter is that the results will be more accurate. The disadvantage is that the algorithm will be slightly slower, however according to our performance tests, this change has only a marginal impact on the run time of the tool.

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)
mglimOneGroup	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

First, find the **Empirical analysis of DGE** tool:

Toolbox | Expression Analysis (📁) | Statistical Analysis | Empirical Analysis of DGE


The original count data for a full expression experiment are the expected input to the Empirical Analysis of DGE tool.

When Experiments created within the Workbench are used as input, the original count values are always used. Columns of such Experiments that contain transformed or normalized values are ignored.

If expression values are being imported from outside the Workbench for use with this test, the data should be original (non-transformed, non-normalized) counts.

Whether the data has been generated in the Workbench or outside the Workbench and imported, the full set of expression results should be used. Please do not run this test on a subset of values from the original sample data.

The reason that the complete set of original count data for samples should be used as input to this test is that the algorithm assumes that the counts on which it operates are Negative Binomially distributed. It implicitly normalizes and transforms these counts, so if the counts have 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 22.39). 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 are low are likely to contribute mostly with noise to the estimation, and features with a lower cumulative 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.

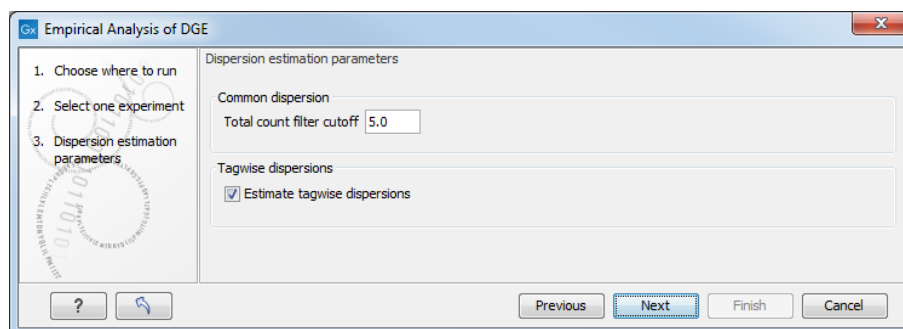


Figure 22.39: Empirical analysis of DGE: setting the parameters related to dispersion.

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 22.40). 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). For example, the All pairs option should be selected when you wish to perform the test of equality for group means for all of the pairs, e.g. if you would like to compare different tissues where each tissue is represented in a group. In this case there is no reference group, so the following comparisons will be performed:

- liver vs heart
- liver vs lung
- heart vs lung

The Against reference option should be selected when you wish to perform the test of equality for group means against one group, the reference, rather than all groups as above. Against reference could be used if you have a wild type and some mutant groups, e.g. Wild type, Mutant 1 and Mutant 2. In this case you might be interested in comparing the mutants to the wild type, but comparing the mutants to each other is not of interest. In this case the Wild Type group is considered the reference and the comparisons will be performed:

- Wild type vs Mutant 1
- Wild type vs Mutant 2

Note that with the Against reference option fewer comparisons are made, as in the above example where Mutant 1 vs. Mutant 2 is not considered.

Below you can select to add two kinds of corrected p-values to the analysis (in addition to the standard p-value produced for the test statistic):

- **Bonferroni corrected.**
- **FDR corrected.**

Both are calculated from the original p-values, and aim in different ways to take into account the issue of multiple testing [Dudoit et al., 2003]. The problem of multiple testing arises because the original p-values are related to a single test: the p-value is the probability of observing a more extreme value than that observed in the test carried out. If the p-value is 0.04, we would expect an as extreme value as that observed in 4 out of 100 tests carried out among groups with no difference in means. Popularly speaking, if we carry out 10000 tests and select the features with original p-values below 0.05, we will expect about 0.05 times 10000 = 500 to be false positives.

The Bonferroni corrected p-values handle the multiple testing problem by controlling the 'family-wise error rate': the probability of making at least one false positive call. They are calculated by multiplying the original p-values by the number of tests performed. The probability of having at least one false positive among the set of features with Bonferroni corrected p-values below 0.05, is less than 5%. The Bonferroni correction is conservative: there may be many genes that are differentially expressed among the genes with Bonferroni corrected p-values above 0.05, that will be missed if this correction is applied.

Instead of controlling the family-wise error rate we can control the false discovery rate: FDR. The false discovery rate is the proportion of false positives among all those declared positive. We expect 5 % of the features with FDR corrected p-values below 0.05 to be false positive. There are many methods for controlling the FDR - the method used in *CLC Main Workbench* is that of [Benjamini and Hochberg, 1995].

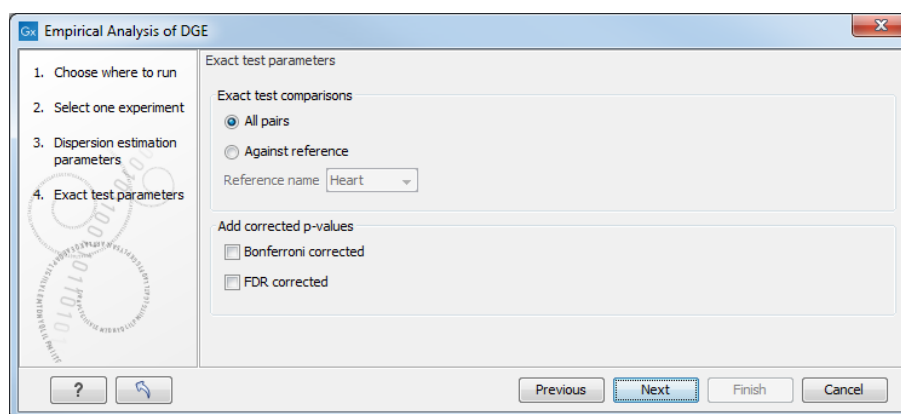


Figure 22.40: Empirical analysis of DGE: setting comparisons and corrected p-value options.

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 relative abundances, which are derived internally in the Exact Test algorithm. They depend on both the sizes (depth of coverage/library size) 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 relative abundance of group 2 is relative to that of group 1. If the relative abundance of group 2 is bigger than that of group 1 the fold change is the relative abundance of group 2 divided by that of group 1. If the relative abundance of group 2 is smaller than that of group 1 the fold change is the relative abundance of group 1 divided by that of group 2 with a negative sign. The 'weighted difference' column contains the difference between the relative abundance of group 2 and the relative abundance 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.

22.4.2 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 tag profiling for example. 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 22.1.1), 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 specify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

Note that the proportion-based tests use the total sample counts (that is, the sum over all expression values). If one (or more) of the counts are NaN, the sum will be NaN and all the test statistics will be NaN. As a consequence all p-values will also be NaN. You can avoid this by filtering your experiment and creating a new experiment so that no NaN values are present, before you apply the tests.

Kal et al.'s test (Z-test)

Kal et al.'s test [Kal et al., 1999] compares a single sample against another single sample, and thus requires that each group in you experiment has only one sample. The test relies on an approximation of the binomial distribution by the normal distribution [Kal et al., 1999]. Considering proportions rather than raw counts the test is also suitable in situations where the sum of counts is different between the samples.

When Kal's test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Proportions difference' column contains the difference between the proportion in group 2 and the proportion in group 1. The 'Fold Change' column tells you how many times bigger the proportion in group 2 is relative to that of group 1. If the proportion in group 2 is bigger than that in group 1 this value is the proportion in group 2 divided by that in group 1. If the proportion in group 2 is smaller than that in group 1 the fold change is the proportion in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR

corrected p-values were chosen.

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.

22.4.3 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 22.41.

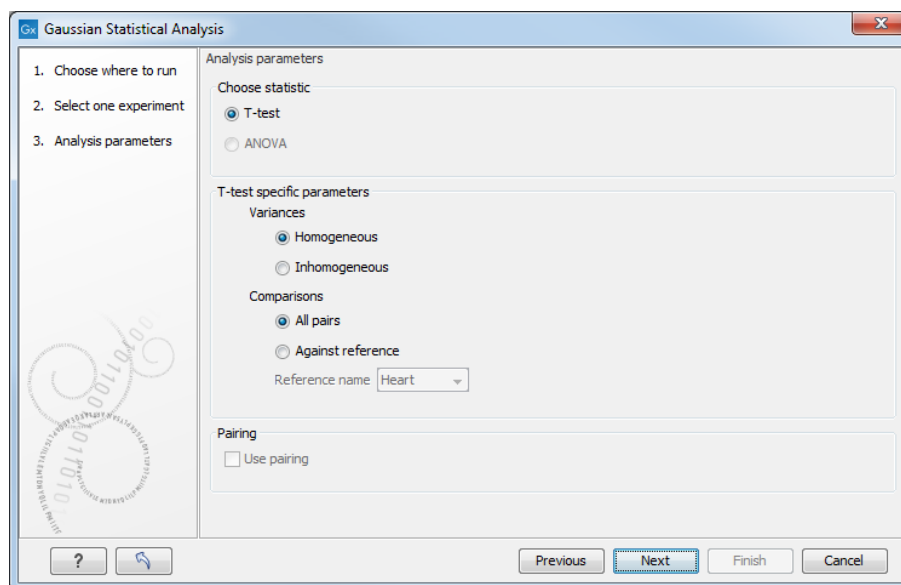


Figure 22.41: Selecting a t-test.

There are different types of t-tests, depending on the assumption you make about the variances in the groups. By selecting 'Homogeneous' (the default) calculations are done assuming that the groups have equal variances. When 'In-homogeneous' is selected, this assumption is not made.

The t-test can also be chosen if you have a multi-group experiment. In this case you may choose either to have t-tests produced for all pairs of groups (by clicking the 'All pairs' button) or to have a t-test produced for each group compared to a specified reference group (by clicking the 'Against reference' button). In the last case you must specify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

If a experiment with pairing was set up (see section 22.1.1) 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.

ANOVA

For experiments with more than two groups you can choose **T-test** as described above, or **ANOVA**.

The ANOVA method allows analysis of an experiment with one factor and a number of groups, e.g. different types of tissues, or time points. In the analysis, the variance within groups is compared to the variance between groups. You get a significant result (that is, a small ANOVA p-value) if the difference you see between groups relative to that within groups, is larger than what you would expect, if the data were really drawn from groups with equal means.

If an experiment with pairing was set up (see section 22.1.1) 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.

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.

22.4.4 Corrected p-values

Clicking **Next** will display a dialog as shown in figure 22.42.

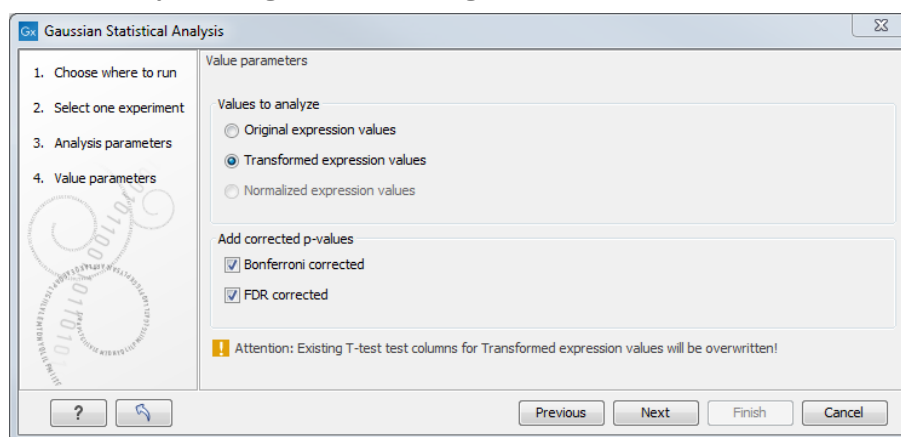


Figure 22.42: Additional settings for the statistical analysis.

At the top, you can select which values to analyze (see section 22.2.1).

Below you can select to add two kinds of corrected p-values to the analysis (in addition to the standard p-value produced for the test statistic):

- **Bonferroni corrected.**
- **FDR corrected.**

Both are calculated from the original p-values, and aim in different ways to take into account the issue of multiple testing [Dudoit et al., 2003]. The problem of multiple testing arises because the original p-values are related to a single test: the p-value is the probability of observing a more extreme value than that observed in the test carried out. If the p-value is 0.04, we would expect an as extreme value as that observed in 4 out of 100 tests carried out among groups with no difference in means. Popularly speaking, if we carry out 10000 tests and select the features with original p-values below 0.05, we will expect about 0.05 times 10000 = 500 to be false positives.

The Bonferroni corrected p-values handle the multiple testing problem by controlling the 'family-wise error rate': the probability of making at least one false positive call. They are calculated by multiplying the original p-values by the number of tests performed. The probability of having at least one false positive among the set of features with Bonferroni corrected p-values below 0.05, is less than 5%. The Bonferroni correction is conservative: there may be many genes that are differentially expressed among the genes with Bonferroni corrected p-values above 0.05, that will be missed if this correction is applied.


Instead of controlling the family-wise error rate we can control the false discovery rate: FDR. The false discovery rate is the proportion of false positives among all those declared positive. We expect 5 % of the features with FDR corrected p-values below 0.05 to be false positive. There are many methods for controlling the FDR - the method used in *CLC Main Workbench* is that of [Benjamini and Hochberg, 1995].

Click **Finish** to start the tool.

Note that if you have already performed statistical analysis on the same values, the existing one will be overwritten.

22.4.5 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 22.1.2). 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 3.3).

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 22.1.4, 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 22.43.

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

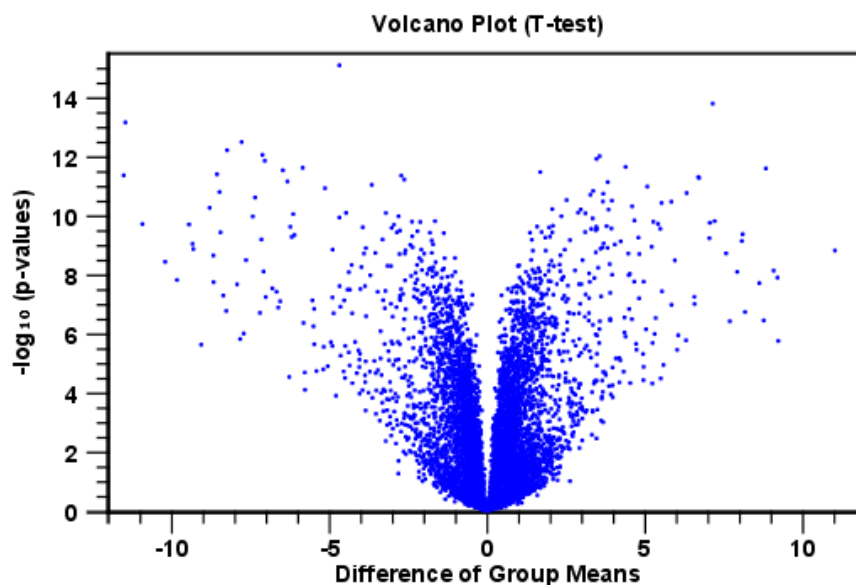


Figure 22.43: *Volcano plot.*

the scale of your data (Read the note on fold change in section [22.1.2](#)).

The larger the difference in expression of a feature, the more extreme its 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 [2.2](#)).

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 [22.1.2](#).

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 [4.6](#)).

22.5 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.

22.5.1 Hierarchical clustering of features

A hierarchical clustering of features is a tree presentation of the similarity in expression profiles of the features over a set of samples (or groups).

The tree structure is generated by

1. letting each feature be a cluster
2. calculating pairwise distances between all clusters
3. joining the two closest clusters into one new cluster
4. iterating 2-3 until there is only one cluster left (which will contain all samples).

The tree is drawn so that the distances between clusters are reflected by the lengths of the branches in the tree. Thus, features with expression profiles that closely resemble each other have short distances between them, those that are more different, are placed further apart.

To start the clustering of features:

Toolbox | Expression Analysis (🇺🇸) | Feature Clustering | Hierarchical Clustering of Features (🇺🇸)

Select at least two samples (🇺🇸 or 🇺🇸) or an experiment (🇺🇸).

Note! If your data contains many features, the clustering will take very long time and could make your computer unresponsive. It is recommended to perform this analysis on a subset of the data (which also makes it easier to make sense of the clustering. Typically, you will want to filter away the features that are thought to represent only noise, e.g. those with mostly low values, or with little difference between the samples). See how to create a sub-experiment in section 22.1.2.

Clicking **Next** will display a dialog as shown in figure 22.44. 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.

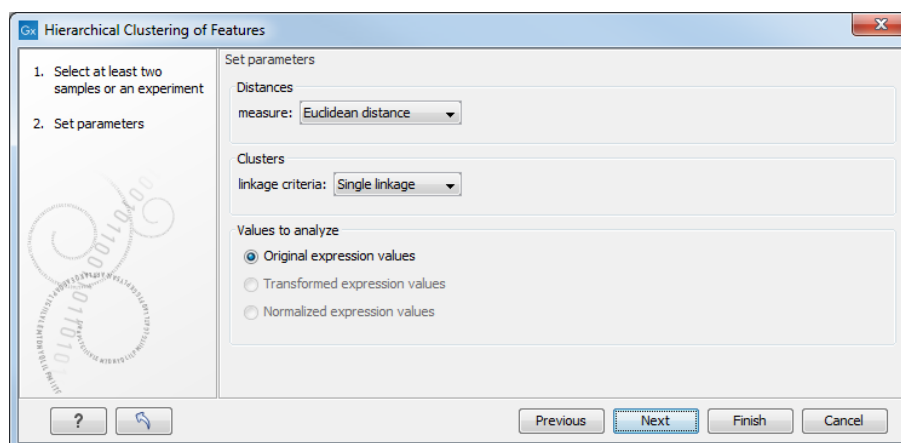


Figure 22.44: Parameters for hierarchical clustering of features.

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 $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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 $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$ is defined as

$$r = \frac{1}{n-1} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s_x} \right) * \left(\frac{y_i - \bar{y}}{s_y} \right)$$

where \bar{x}/\bar{y} is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance.** The Manhattan distance between two points is the distance measured along axes at right angles. If $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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 22.2.1). Click **Finish** to start the tool.

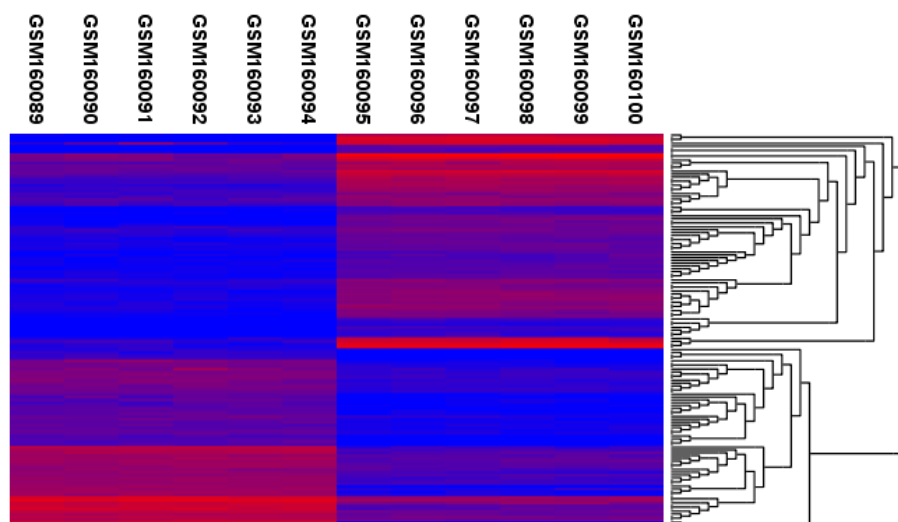


Figure 22.45: Hierarchical clustering of features.



Figure 22.46: Showing the hierarchical clustering of an experiment.

Result of hierarchical clustering of features

The result of a feature clustering is shown in figure 22.45.

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 22.46).

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 22.45). 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 22.47).

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 22.48).

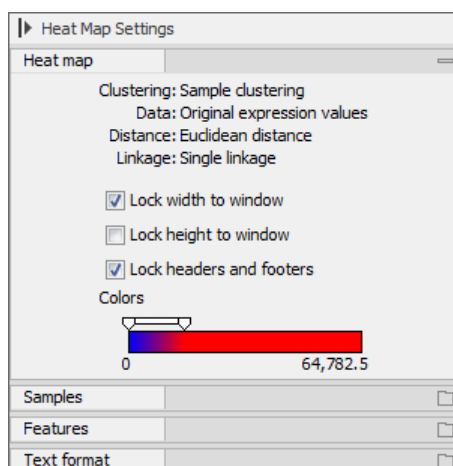


Figure 22.47: Side Panel of heat map.

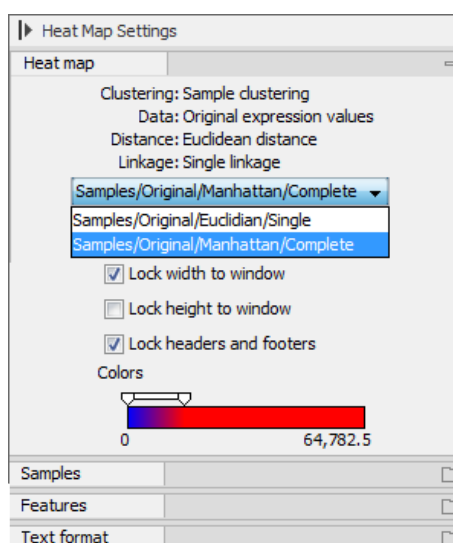


Figure 22.48: When more than one clustering has been performed, there will be a list of heat maps to choose from.

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 4.6).

22.5.2 K-means/medoids clustering

In a k-means or medoids clustering, features are clustered into k separate clusters. The procedures seek to find an assignment of features to clusters, for which the distances between features within the cluster is small, while distances between clusters are large.

Toolbox | Expression Analysis (🇺🇸) | Feature Clustering | K-means/medoids Clustering (🇺🇸)

Select at least two samples (🇺🇸 or 🇺🇸) or an experiment (🇺🇸).

Note! If your data contains many features, the clustering will take very long time and could make your computer unresponsive. It is recommended to perform this analysis on a subset of the data (which also makes it easier to make sense of the clustering). See how to create a sub-experiment in section 22.1.2.

Clicking **Next** will display a dialog as shown in figure 22.49.

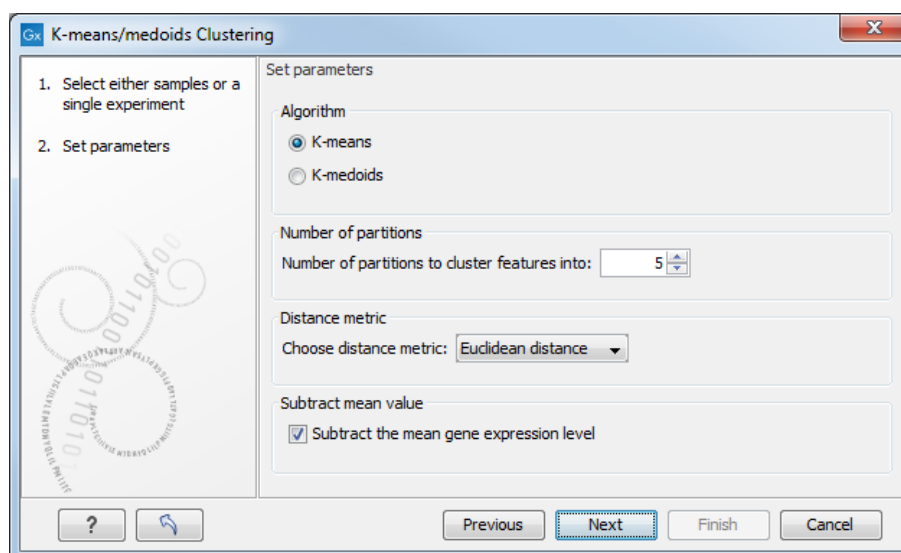


Figure 22.49: Parameters for k-means/medoids clustering.

The parameters are:

- **Algorithm.** You can choose between two clustering methods:

- **K-means.** K-means clustering assigns each point to the cluster whose center is nearest. The center/centroid of a cluster is defined as the average of all points in the cluster. If a data set has three dimensions and the cluster has two points $X = (x_1, x_2, x_3)$ and $Y = (y_1, y_2, y_3)$, then the centroid Z becomes $Z = (z_1, z_2, z_3)$, where $z_i = (x_i + y_i)/2$ for $i = 1, 2, 3$. The algorithm attempts to minimize the intra-cluster variance defined by:

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

where there are k clusters $S_i, i = 1, 2, \dots, k$ 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 $S_i, i = 1, 2, \dots, k$ 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-algorithm is described in [Kaufman and Rousseeuw, 1990].

- **Number of partitions.** The maximum number of partitions to cluster features into: the final number of clusters can be smaller than that.
- **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 $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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 $u = (u_1, u_2, \dots, u_n)$ and $v = (v_1, v_2, \dots, v_n)$, 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.

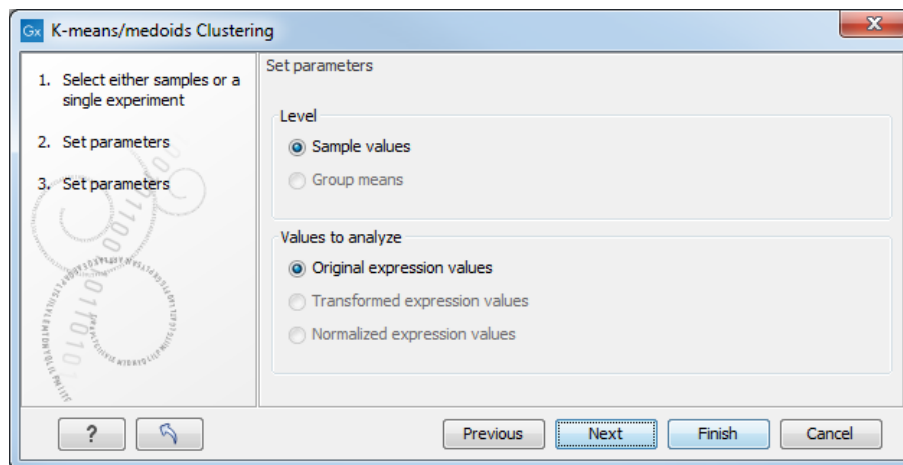


Figure 22.50: Parameters for *k*-means/medoids clustering.

Clicking **Next** will display a dialog as shown in figure 22.50.

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 22.2.1).

Click **Finish** to start the tool.

The *k*-means implementation first assigns each feature to a cluster at random. Then, at each iteration, it reassigns features to the centroid of the nearest cluster. During this reassignment, it can happen that one or more of the clusters becomes empty, explaining why the final number of clusters might be smaller than the one specified in "number of partitions". Note that the initial assignment of features to clusters is random, so results can differ when the algorithm is run again.

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 22.49) - there is one graph per cluster. Using drag and drop as explained in section 2.1.6, you can arrange the views to see more than one graph at the time.

Figure 22.51 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 than 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 22.1.5.

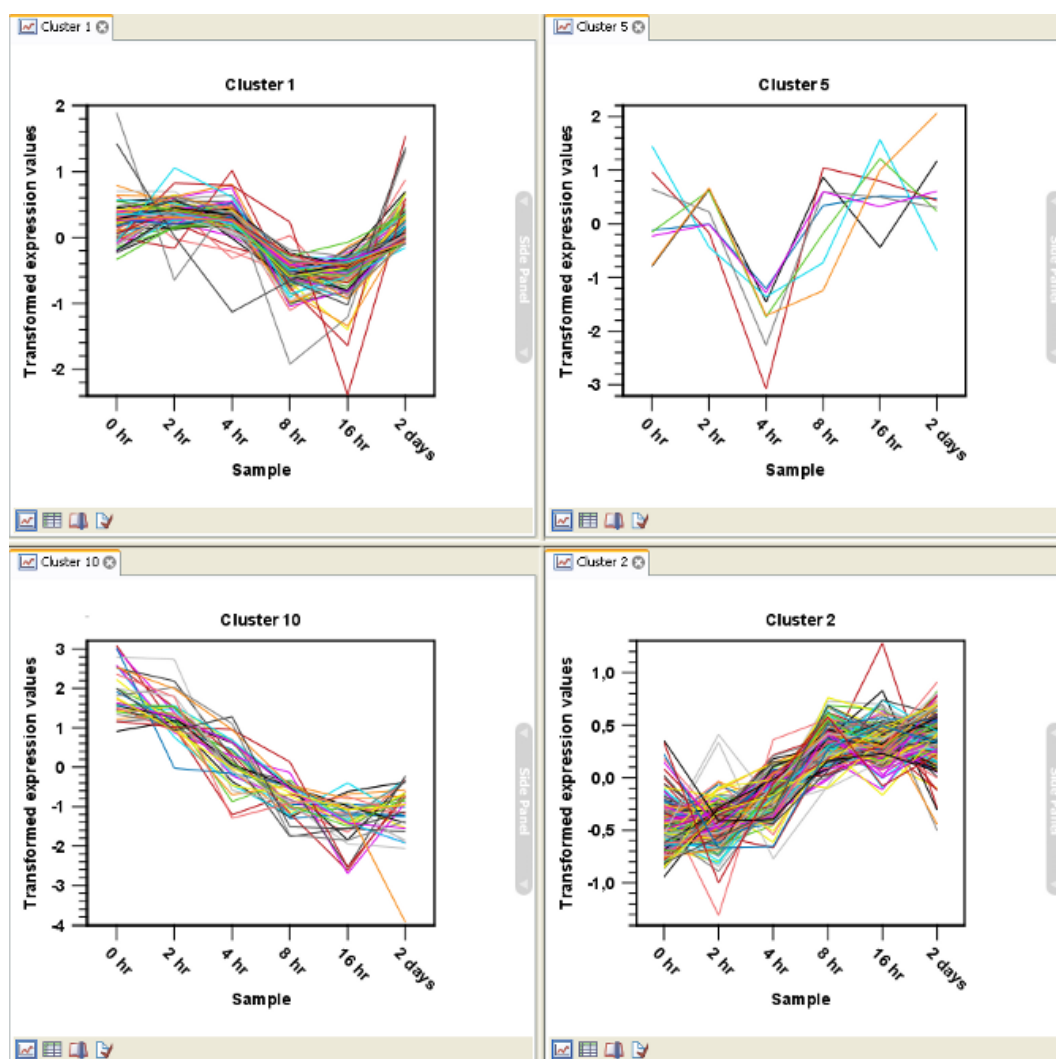


Figure 22.51: Four clusters created by k-means/medoids clustering.

22.6 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 [22.1.3](#).

22.6.1 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 | Expression Analysis (📁) | Annotation Test | Hypergeometric Tests on Annotations (🌐)

This will show a dialog where you can select the two experiments - the larger experiment, e.g. the original experiment including the full list of features - and a sub-experiment (see how to create a sub-experiment in section 22.1.2).

Click **Next**. This will display the dialog shown in figure 22.52.

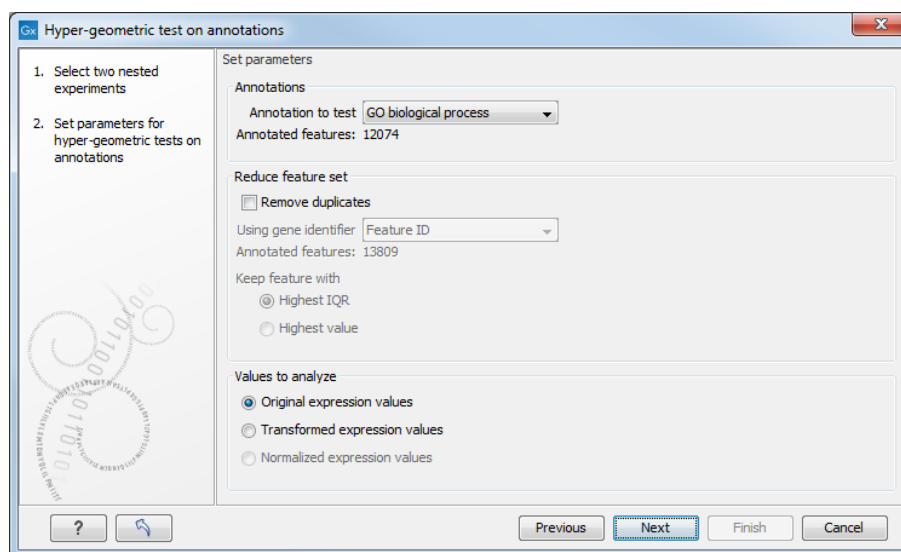


Figure 22.52: Parameters for performing a hypergeometric test on annotations.

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 22.2.1). 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 **Finish** to start the tool.

Result of hypergeometric tests on annotations

The result of performing hypergeometric tests on annotations using GO biological process is shown in figure 22.53.

Category	Description	Full set	In subset	Expected in subset	Observed - expected	p-value
0055114	oxidation-reduction process (MGI:MGI:483...	561	11	3	8	4.59E-4
0051496	positive regulation of stress fiber assembly...	27	3	0	3	5.25E-4
0001913	T cell mediated cytotoxicity (MGI:MGI:303...	7	2	0	2	7.10E-4
0006629	lipid metabolic process (MGI:MGI:1354194 ...	351	8	2	6	1.10E-3
0006956	complement activation (MGI:MGI:3833206]...	9	2	0	2	1.21E-3
0009058	biosynthetic process (MGI:MGI:2152098 [I...	38	3	0	3	1.45E-3
0006855	drug transmembrane transport (MGI:MGI:...	11	2	0	2	1.83E-3
0032869	cellular response to insulin stimulus (MGI:M...	45	3	0	3	2.36E-3
0008152	metabolic process (MGI:MGI:2152098 [IEA...	456	8	3	5	5.51E-3
0000105	histidine biosynthetic process (MGI:MGI:13...	1	1	0	1	5.90E-3
2001213	negative regulation of vasculogenesis (MG...	1	1	0	1	5.90E-3
0048241	epinephrine transport (MGI:MGI:4417868 [...	1	1	0	1	5.90E-3
0009115	xanthine catabolic process (MGI:MGI:4417...	1	1	0	1	5.90E-3
0050427	3'-phosphoadenosine 5'-phosphosulfate m...	1	1	0	1	5.90E-3
0033301	cell cycle comprising mitosis without cytokin...	1	1	0	1	5.90E-3
0006507	GPI anchor release (MGI:MGI:5447609)PM...	1	1	0	1	5.90E-3

Figure 22.53: 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 over-represented on the features in the subset relative to the full set.

22.6.2 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 that 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 of differential expression, prior to running GSEA on it.

Toolbox | Expression Analysis (🇬🇧) | Annotation Test | Gene Set Enrichment Analysis (GSEA) (🇬🇧)

Select an experiment and click **Next**.

Click **Next**. This will display the dialog shown in figure 22.54.

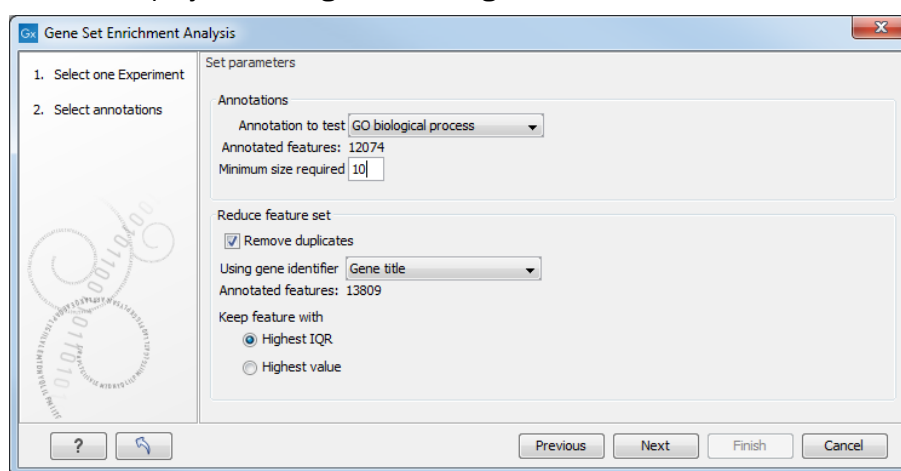


Figure 22.54: Gene set enrichment analysis on GO biological process.

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 22.55.

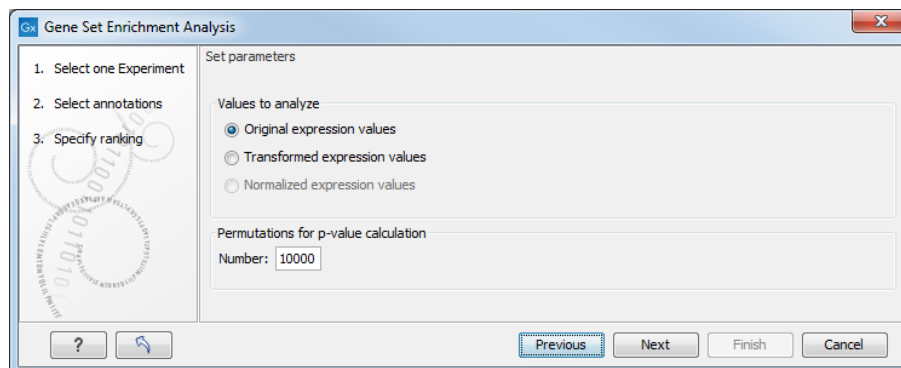


Figure 22.55: Gene set enrichment analysis parameters.

At the top, you can select which values to analyze (see section 22.2.1).

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 **Finish** to start the tool.

Result of gene set enrichment analysis

The result of performing gene set enrichment analysis using GO biological process is shown in figure 22.56.

Category	Description	Size	Test statistic	Lower tail	Upper tail
0006508	proteolysis	658	-11.83	0.00	1.00
0044260	cellular macromolecule metabolic process	3323	-9.70	0.00	1.00
0044264	cellular polysaccharide metabolic process	50	-21.35	0.00	1.00
0048634	regulation of muscle organ development	102	-17.84	0.00	1.00
0048641	regulation of skeletal muscle tissue development	44	-32.11	0.00	1.00
0048642	negative regulation of skeletal muscle tissue development	10	-26.77	0.00	1.00
0014904	myotube cell development	23	-28.90	0.00	1.00
0045661	regulation of myoblast differentiation	39	-20.11	0.00	1.00
0005977	glycogen metabolic process	44	-24.15	0.00	1.00
0014888	striated muscle adaptation	22	-25.33	0.00	1.00
0048741	skeletal muscle fiber development	21	-30.18	0.00	1.00
0014819	regulation of skeletal muscle contraction	10	-34.57	0.00	1.00
0043170	macromolecule metabolic process	3739	-10.33	0.00	1.00
0019538	protein metabolic process	2337	-8.90	0.00	1.00
0007519	skeletal muscle tissue development	52	-25.32	0.00	1.00
0009057	macromolecule catabolic process	450	-13.59	0.00	1.00

Figure 22.56: The result of gene set enrichment analysis on GO biological process.

The table shows the following information:

- **Category.** This is the identifier for the category.
- **Description.** This is the description belonging to the category. Both of these are simply extracted from the annotations.
- **Size.** The number of features with this category. (Note that this is after removal of duplicates).
- **Test statistic.** This is the GSEA test statistic.
- **Lower tail.** This is the mass in the permutation based p-value distribution below the value of the test statistic.
- **Upper tail.** This is the mass in the permutation based p-value distribution above the value of the test statistic.

A small lower (or upper) tail p-value for an annotation category is an indication that features in this category viewed as a whole are perturbed among the groups in the experiment considered.




22.7 General plots

In the **General Plots** folder, you find three general plots that may be useful at various point of your analysis work flow. The plots are explained in detail below.

22.7.1 Histogram

A histogram shows a distribution of a set of values. Histograms are often used for examining and comparing distributions, e.g. of expression values of different samples, in the quality control step of an analysis. You can create a histogram showing the distribution of expression value for a sample:

Toolbox | Expression Analysis () | General Plots | Create Histogram ()

Select a number of samples (), (), () or 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 22.57.

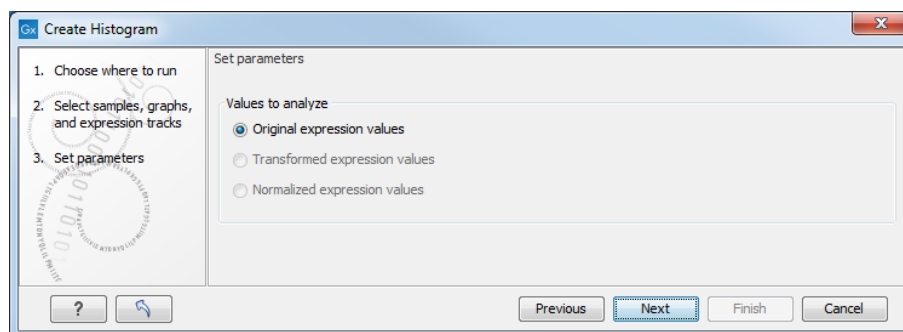


Figure 22.57: Selecting which values the histogram should be based on.

In this dialog, you select the values to be used for creating the histogram (see section 22.2.1).

Click **Finish** to start the tool.

Viewing histograms

The resulting histogram is shown in a figure 22.58

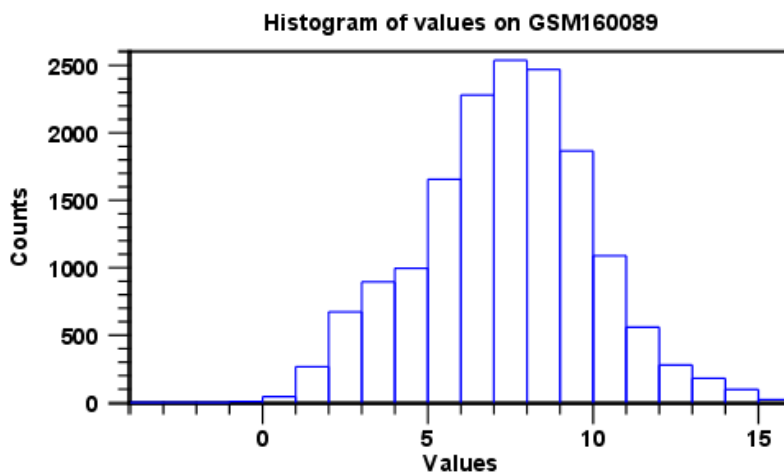


Figure 22.58: Histogram showing the distribution of transformed expression values.

The histogram shows the expression value on the x axis (in the case of figure 22.58 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 4.6).

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 22.59.

Data	Count
Total number of values	15923
Number values used	15923
Number -Inf values	0
Number +Inf values	0
Number NaN values	0

Figure 22.59: 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**

22.7.2 MA plot

The MA plot is a scatter rotated by 45° . For two samples of expression values it plots for each gene the difference in expression against the mean expression level. MA plots are often used for quality control, in particular, to assess whether normalization and/or transformation is required.

You can create an MA plot comparing two samples:

Toolbox | Expression Analysis (📁) | General Plots | Create MA Plot (👉)

In the first two dialogs, select two samples (🇩🇪), (🇺🇸) or (🇺🇸): the first must be the case expression data, and the second the control data. Clicking **Next** will display a dialog as shown in figure 22.60.

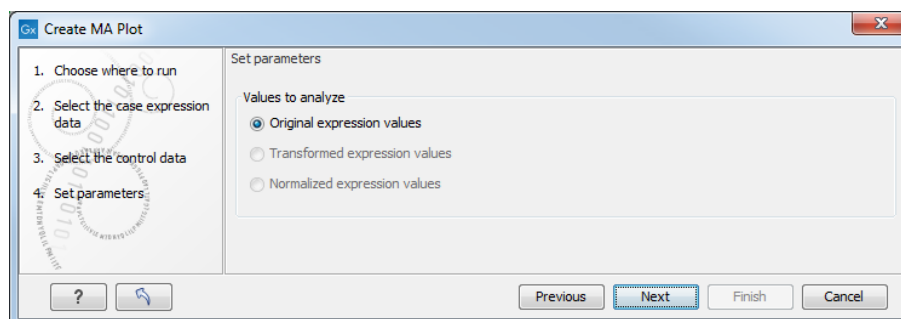


Figure 22.60: Selecting which values the MA plot should be based on.

In this dialog, you select the values to be used for creating the MA plot (see section 22.2.1).

Click **Finish** to start the tool.

Viewing MA plots

The resulting plot is shown in a figure 22.61.

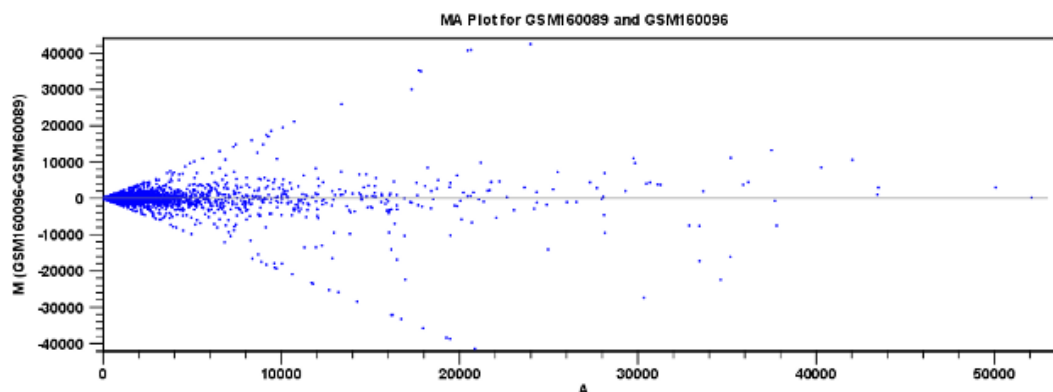


Figure 22.61: MA plot based on original expression values.

The X axis shows the mean expression level of a feature on the two samples and the Y axis shows the difference in expression levels for a feature on the two samples. From the plot shown in figure 22.61 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 22.2.2).

Figure 22.62 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 dependence of the variance on the mean is not as strong as it was before transformation.

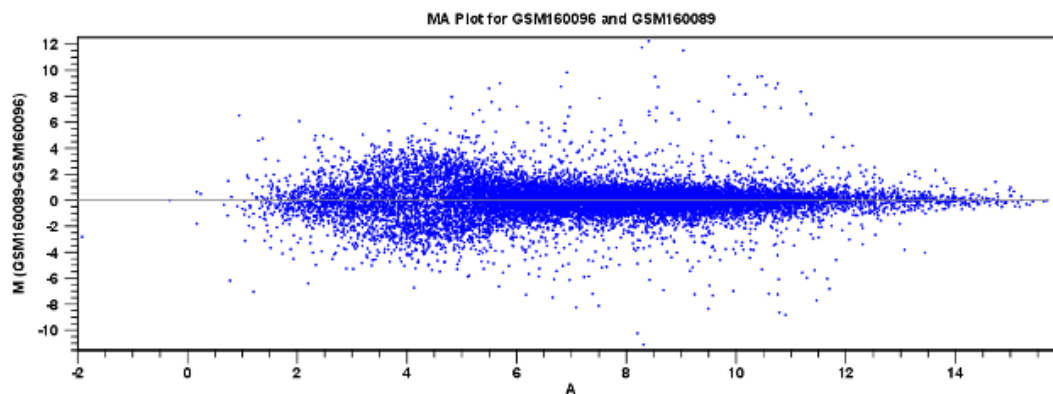


Figure 22.62: MA plot based on transformed expression values.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
 - Outside
 - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
 - None
 - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **y = 0 axis**. Draws a line where $y = 0$. Below there are some options to control the appearance of the line:
 - **Line width**
 - * Thin
 - * Medium
 - * Wide
 - **Line type**

- * None
- * Line
- * Long dash
- * Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

- **Line width**

- Thin
- Medium
- Wide

- **Line type**

- None
- Line
- Long dash
- Short dash

- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties** preferences, where you can adjust coloring and appearance of the dots:

- **Dot type**

- None
- Cross
- Plus
- Square
- Diamond
- Circle
- Triangle
- Reverse triangle
- Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a scatter plot, you need to save the settings of the **Side Panel** (see section 4.6).

22.7.3 Scatter plot

As described in section 22.1.4, an experiment can be viewed as a scatter plot. However, you can also create a "stand-alone" scatter plot of two samples:

Toolbox | Expression Analysis (🇺🇸) | General Plots | Create Scatter Plot (📊)

In the first two dialogs, select two samples (🇺🇸, 🇺🇸) or (🇺🇸, 🇺🇸): the first is the sample that will be plotted on the X axis of the plot, the second the one that will define the Y axis. Clicking **Next** will display a dialog as shown in figure 22.63.

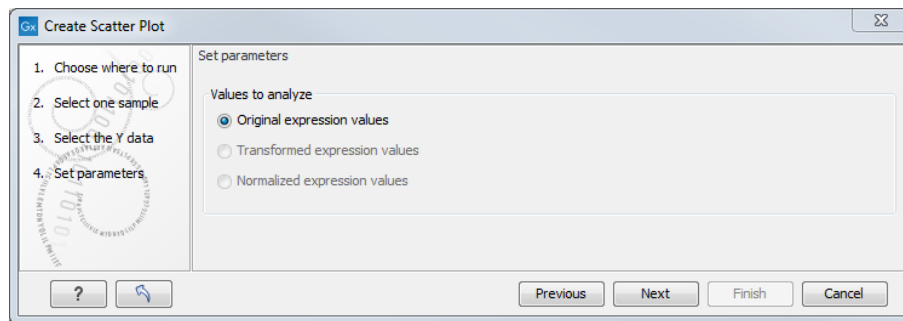


Figure 22.63: Selecting which values the scatter plot should be based on.

In this dialog, you select the values to be used for creating the scatter plot (see section 22.2.1).

Click **Finish** to start the tool.

For more information about the scatter plot view and how to interpret it, please see section 22.1.4.

Chapter 23

BLAST search

Contents

23.1 Running BLAST searches	519
23.1.1 BLAST at NCBI	519
23.1.2 BLAST against local data	522
23.2 Output from BLAST searches	526
23.2.1 Graphical overview for each query sequence	526
23.2.2 Overview BLAST table	527
23.2.3 BLAST graphics	528
23.2.4 BLAST HSP table	529
23.2.5 BLAST hit table	531
23.3 Extract consensus sequence	532
23.4 Local BLAST databases	534
23.4.1 Make pre-formatted BLAST databases available	534
23.4.2 Download NCBI pre-formatted BLAST databases	535
23.4.3 Create local BLAST databases	536
23.5 Manage BLAST databases	537
23.6 Bioinformatics explained: BLAST	538
23.6.1 How does BLAST work?	539
23.6.2 Which BLAST program should I use?	540
23.6.3 Which BLAST options should I change?	542
23.6.4 Explanation of the BLAST output	543
23.6.5 I want to BLAST against my own sequence database, is this possible?	544
23.6.6 What you cannot get out of BLAST	546
23.6.7 Other useful resources	546

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 23.6.

Figure 23.8 shows an example of a BLAST result in the *CLC Main Workbench*.

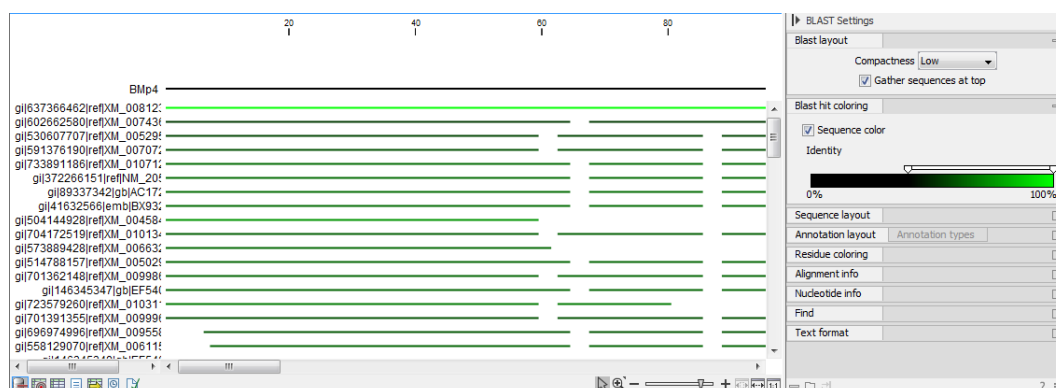


Figure 23.1: Display of the output of a BLAST search. At the top is there a graphical representation of BLAST hits with tool-tips showing additional information on individual hits. Below is a tabular form of the BLAST results.

23.1 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.

23.1.1 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, go to:

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 23.2

Select one or more sequences of the same type (either DNA or protein) and click **Next**.

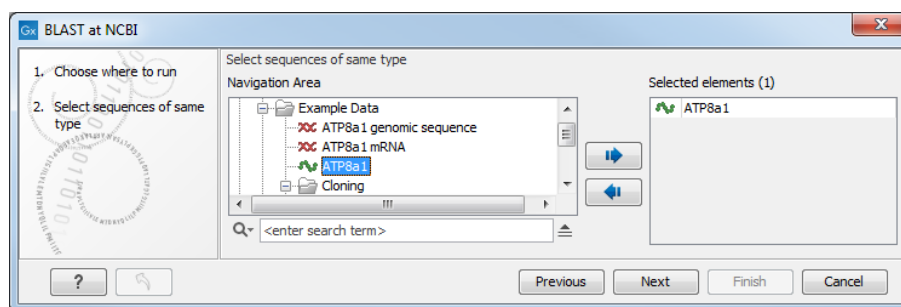


Figure 23.2: Choose one or more sequences to conduct a BLAST search with.

In this dialog, you choose which type of BLAST search to conduct, and which database to search against (figure 23.3). 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 choose among to run. A complete list of these databases can be found in Appendix B. Here you can also read how to add additional databases available the NCBI to the list provided in the dropdown menu.

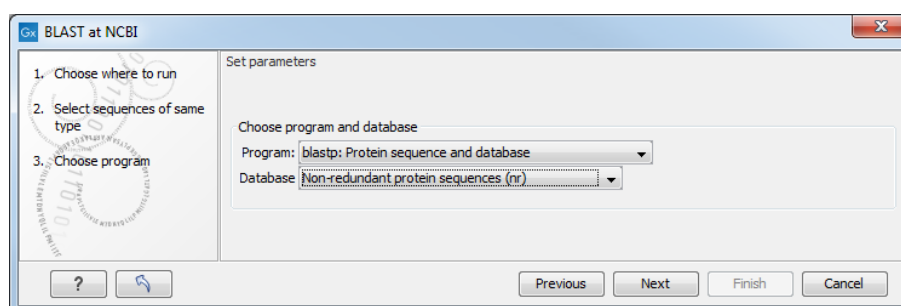


Figure 23.3: Choose a BLAST Program and a database for the search.

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 23.4, allows you to choose parameters to tune your BLAST search, to meet your requirements.

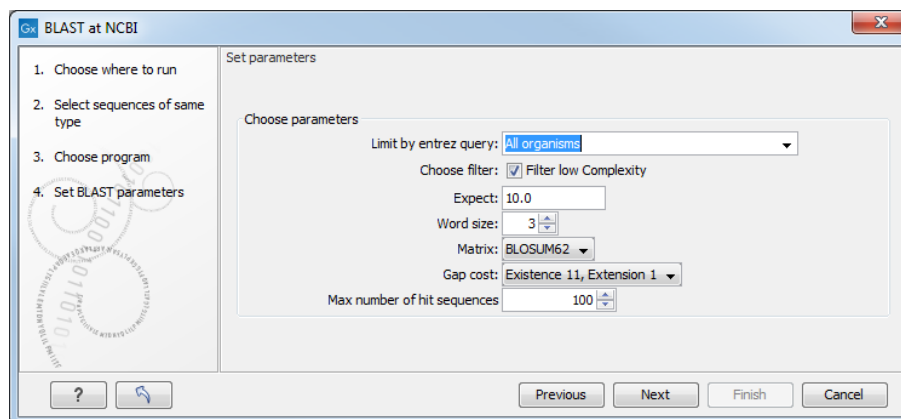


Figure 23.4: Parameters that can be set before submitting a BLAST search.

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/blastcguihelp.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 **Finish** to start the tool.

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 23.3 and the rest of the options are the same as when performing a BLAST search with a full sequence.

23.1.2 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, *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 23.4.3)
- You can add pre-formatted databases (see section 23.4.1)
- You can use sequence data from the **Navigation Area** directly, without creating a database first.

To conduct a local BLAST search, go to:

Toolbox | BLAST (📁) | **BLAST** (📁)

This opens the dialog seen in figure 23.5:

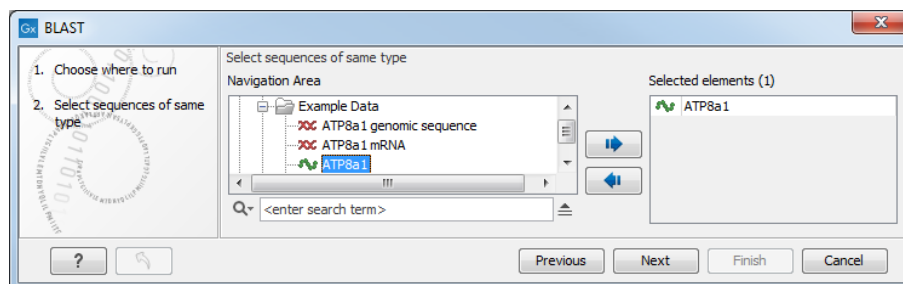


Figure 23.5: Choose one or more sequences to conduct a BLAST search.

Select one or more sequences of the same type (DNA or protein) and click **Next**.

This opens the dialog seen in figure 23.6:

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.
- **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.

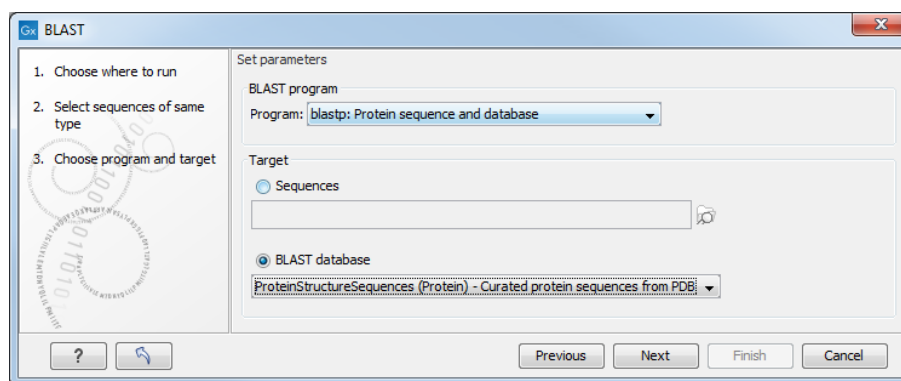


Figure 23.6: Choose a BLAST program and a target database.

- **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 [12.1.3](#)).

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 [23.4.3](#).
- **BLAST Database.** Select a database already available in one of your designated BLAST database folders. Read more in section [23.5](#).

When a database or a set of sequences has been selected, click **Next**.

The next dialog allows you to adjust the parameters to meet the requirements of your BLAST search (figure [23.7](#)).

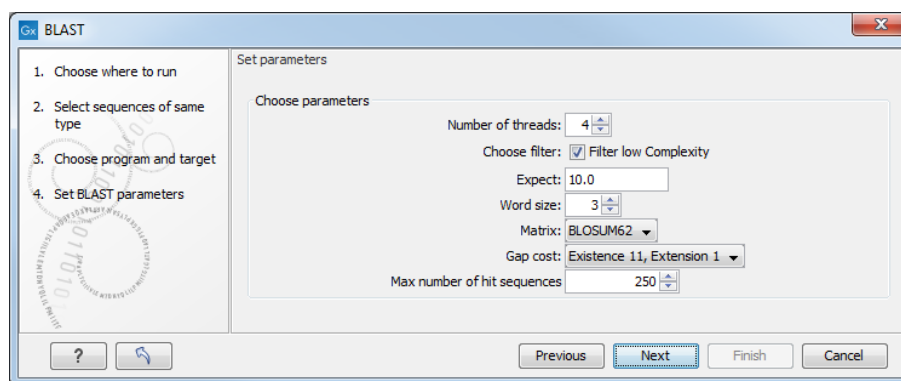


Figure 23.7: Parameters that can be set before submitting a local BLAST search.

- **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 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.

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 23.6 and the rest of the options are the same as when performing a BLAST search with a full sequence.

23.2 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 and High-Scoring Segment Pairs (HSPs) 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 23.2.2.

23.2.1 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 23.8. In cases where only one sequence was entered into a BLAST search, such a graphical overview is the default output.

Figure 23.8 shows an example of a BLAST result for an individual query sequence in the *CLC Main Workbench*.

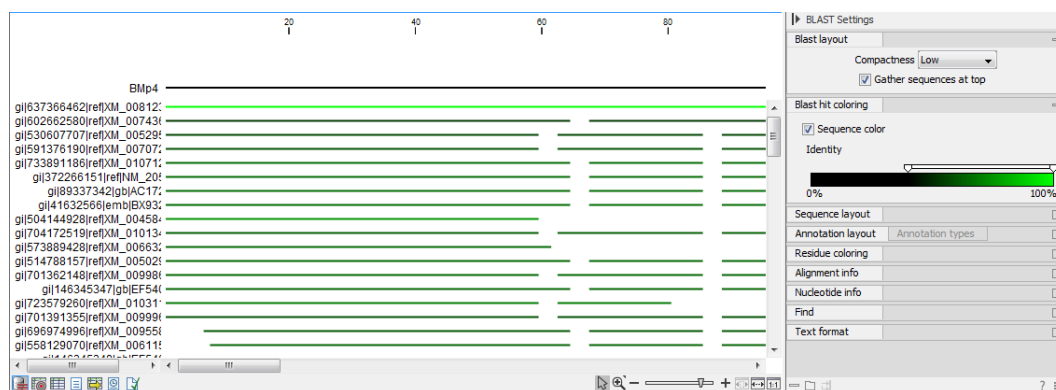


Figure 23.8: Default display of the output of a BLAST search for one query sequence. At the top is there a graphical representation of BLAST hits with tooltips showing additional information on individual hits.

Detailed descriptions of the overview BLAST table and the graphical BLAST results view are described below.

23.2.2 Overview BLAST table

In the overview BLAST table for a multi-sequence blast search, as shown in figure 23.9, there is one row for each query sequence. Each row represents the BLAST result for this query sequence.

The screenshot shows a web-based interface for a BLAST search. At the top, it says 'Rows: 2' and has a 'Filter' button. Below this is a table with the following data:

Query	Number of hits	Lowest E-value	Accession (E-value)
ATP8a1 genomic sequence		624	0.00 ATP8a1_genomic_sequence
ATP8a1 mRNA		50	0.00 ATP8a1_genomic_sequence

At the bottom of the table, there are three buttons: 'Open BLAST Output', 'Extract Consensus', and 'Open Query Sequence'.

Figure 23.9: An overview BLAST table summarizing the results for a number of query sequences.

Double-clicking a row will open the BLAST result for this query sequence, allowing more detailed investigation of the result. You can also select one or more rows and click the **Open BLAST Output** button at the bottom of the view. 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 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 HSPs: The number of High-scoring Segment Pairs (HSPs) for this query sequence.
- For the following list, the value of the best HSP is displayed together with accession number and description of this HSP, 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 HSPs length
 - Accession (HSP length)

- Description (HSP 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.

23.2.3 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.

- **Blast layout.** You can control the level of **Compactness** for displaying sequences:
 - **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.

You can also 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 [11.1](#).

Some of the information available in the tooltips when hovering over a particular hit sequence 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.

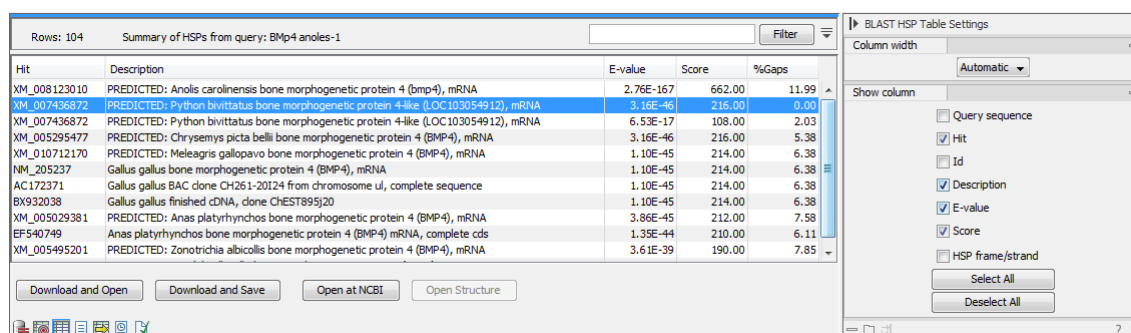
23.2.4 BLAST HSP table

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 HSP Table** button () at the bottom of the view

Figure 23.10 is an example of a BLAST HSP Table.



Hit	Description	E-value	Score	%Gaps
XM_008123010	PREDICTED: Anolis carolinensis bone morphogenetic protein 4 (bmp4), mRNA	2.76E-167	662.00	11.99
XM_007436872	PREDICTED: Python bivittatus bone morphogenetic protein 4-like (LOC103054912), mRNA	3.16E-46	216.00	0.00
XM_007436872	PREDICTED: Python bivittatus bone morphogenetic protein 4-like (LOC103054912), mRNA	6.53E-17	108.00	2.03
XM_005295477	PREDICTED: Chrysemys picta bellii bone morphogenetic protein 4 (BMP4), mRNA	3.16E-46	216.00	5.38
XM_010712170	PREDICTED: Meleagris gallopavo bone morphogenetic protein 4 (BMP4), mRNA	1.10E-45	214.00	6.38
NM_205237	Gallus gallus bone morphogenetic protein 4 (BMP4), mRNA	1.10E-45	214.00	6.38
AC172371	Gallus gallus BAC clone CH261-20124 from chromosome <i>u</i> , complete sequence	1.10E-45	214.00	6.38
BX932038	Gallus gallus finished cDNA, clone CHEST895j20	1.10E-45	214.00	6.38
XM_005029381	PREDICTED: Anas platyrhynchos bone morphogenetic protein 4 (BMP4), mRNA	3.86E-45	212.00	7.58
EF540749	Anas platyrhynchos bone morphogenetic protein 4 (BMP4) mRNA, complete cds	1.35E-44	210.00	6.11
XM_005495201	PREDICTED: Zonotrichia albicollis bone morphogenetic protein 4 (BMP4), mRNA	3.61E-39	190.00	7.85

Figure 23.10: *BLAST HSP Table*. The HSPs can be sorted by the different columns, simply by clicking the column heading.

The BLAST HSP Table includes the following information:

- **Query sequence.** The sequence which was used for the search.
- **HSP.** 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.
- **HSP start.** Shows the start position in the HSP sequence.
- **HSP end.** Shows the end position in the HSP sequence.
- **HSP length.** The length of the HSP.
- **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 HSP 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 HSP sequence.
- **%Identity.** Shows the percentage of identical residues in the query and HSP sequence.
- **Positive.** Shows the number of similar but not necessarily identical residues in the query and HSP sequence.
- **%Positive.** Shows the percentage of similar but not necessarily identical residues in the query and HSP sequence.
- **Gaps.** Shows the number of gaps in the query and HSP sequence.
- **%Gaps.** Shows the percentage of gaps in the query and HSP sequence.
- **Query Frame/Strand.** Shows the frame or strand of the query sequence.
- **HSP Frame/Strand.** Shows the frame or strand of the HSP sequence.

In the **BLAST table** view you can handle the HSP 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 HSP 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 HSPs 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.

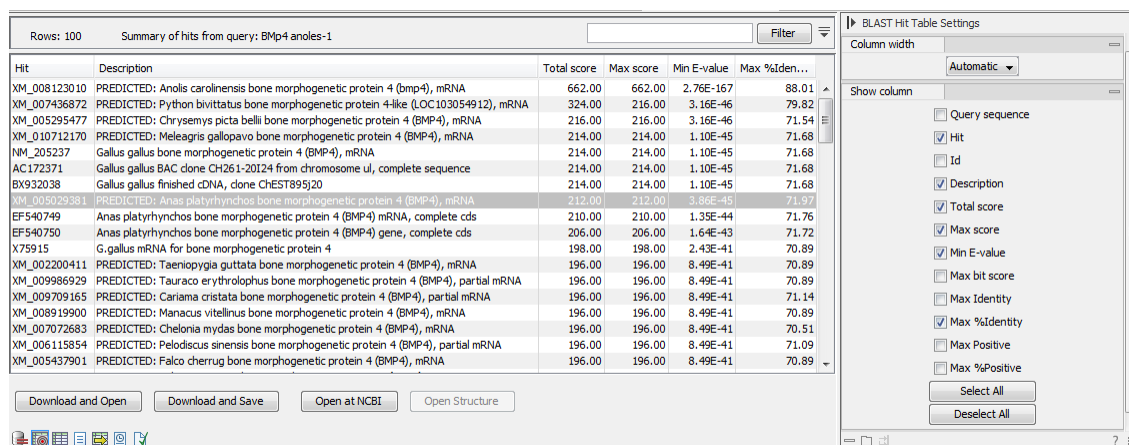
The table is integrated with the graphical view described in section 23.2.3 so that selecting a HSP in the table will make a selection on the corresponding sequence in the graphical view.

23.2.5 BLAST hit table

The **BLAST Hit table** view can be shown in the following way:

Click the **Show BLAST Hit Table** button  at the bottom of the view

Figure 23.11 is an example of a BLAST Hit Table.



Hit	Description	Total score	Max score	Min E-value	Max %Iden...
XM_008123010	PREDICTED: Anolis carolinensis bone morphogenetic protein 4 (bmp4), mRNA	662.00	662.00	2.76E-167	88.01
XM_007436872	PREDICTED: Python bivittatus bone morphogenetic protein 4-like (LOC103054912), mRNA	324.00	216.00	3.16E-46	79.82
XM_005295477	PREDICTED: Chrysemys picta bellii bone morphogenetic protein 4 (BMP4), mRNA	216.00	216.00	3.16E-46	71.54
XM_010712170	PREDICTED: Meleagris gallopavo bone morphogenetic protein 4 (BMP4), mRNA	214.00	214.00	1.10E-45	71.68
NM_205237	Gallus gallus bone morphogenetic protein 4 (BMP4), mRNA	214.00	214.00	1.10E-45	71.68
AC172371	Gallus gallus BAC clone CH261-20124 from chromosome ul, complete sequence	214.00	214.00	1.10E-45	71.68
BX932038	Gallus gallus finished cDNA, done CHEST895J20	214.00	214.00	1.10E-45	71.68
XM_005029381	PREDICTED: Anas platyrhynchos bone morphogenetic protein 4 (BMP4), mRNA	212.00	212.00	3.86E-45	71.97
EF540749	Anas platyrhynchos bone morphogenetic protein 4 (BMP4) mRNA, complete cds	210.00	210.00	1.35E-44	71.76
EF540750	Anas platyrhynchos bone morphogenetic protein 4 (BMP4) gene, complete cds	206.00	206.00	1.64E-43	71.72
X75915	G.gallus mRNA for bone morphogenetic protein 4	198.00	198.00	2.43E-41	70.89
XM_002200411	PREDICTED: Taeniopygia guttata bone morphogenetic protein 4 (BMP4), mRNA	196.00	196.00	8.49E-41	70.89
XM_009986929	PREDICTED: Tauraco erythrophus bone morphogenetic protein 4 (BMP4), partial mRNA	196.00	196.00	8.49E-41	70.89
XM_009709165	PREDICTED: Cariamia cristata bone morphogenetic protein 4 (BMP4), partial mRNA	196.00	196.00	8.49E-41	71.14
XM_008919900	PREDICTED: Manacus vitellinus bone morphogenetic protein 4 (BMP4), mRNA	196.00	196.00	8.49E-41	70.89
XM_007072683	PREDICTED: Chelonia mydas bone morphogenetic protein 4 (BMP4), mRNA	196.00	196.00	8.49E-41	70.51
XM_006115854	PREDICTED: Pelodiscus sinensis bone morphogenetic protein 4 (BMP4), partial mRNA	196.00	196.00	8.49E-41	71.09
XM_005437901	PREDICTED: Falco cherrug bone morphogenetic protein 4 (BMP4), mRNA	196.00	196.00	8.49E-41	70.89

Figure 23.11: *BLAST Hit Table*. The hits can be sorted by the different columns, simply by clicking the column heading.

The BLAST Hit 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.
- **Total Score.** Total score for all HSPs.
- **Max Score.** Maximum score of all HSPs.
- **Min E-value.** Minimum e-value of all HSPs.

- **Max Bit score.** Maximum Bit score of all HSPs.
- **Max Identity.** Shows the maximum number of identical residues in the query and Hit sequence.
- **Max %Identity.** Shows the percentage of maximum identical residues in the query and Hit sequence.
- **Max Positive.** Shows the maximum number of similar but not necessarily identical residues in the query and Hit sequence.
- **Max %Positive.** Shows the percentage of maximum similar but not necessarily identical residues in the query and Hit sequence.

23.3 Extract consensus sequence

You can extract a consensus sequence from a BLAST result. Clicking on the button Extract Consensus Sequence opens a dialog where you can decide how to handle regions with low coverage. The first step is to define a **threshold for when coverage is considered low**. The default value is 0, which means that low coverage is defined as no coverage (i.e. no reads align to the reference at this position). That means if you have one read covering a given position, it will only be that read that determines the consensus sequence. If you need more confidence that the consensus sequence is correct, we advise raising this value. Setting a higher low coverage threshold will require more mapped reads to construct the consensus sequence.

A consensus based on mapped reads cannot be generated in regions that meet or are below the value set for the low coverage threshold, there are several options for handling these low coverage regions:

- **Remove regions with low coverage.** When using this option, no consensus sequence is created for the low coverage regions. There are two ways of creating the consensus sequence from the remaining contiguous stretches of high coverage: either the consensus sequence is **split** into separate sequence when there is a low coverage region, or the low coverage region is simply ignored, and the high-coverage regions are directly **joined** (in this case, an annotation is added at the position where a low coverage region is removed in the consensus sequence produced, see below).
- **Insert 'N' ambiguity symbols.** This will simply add Ns for each base in the low coverage region. An annotation is added for the low coverage region in the consensus sequence produced (see below).
- **Fill from reference sequence.** This option will use the sequence from the reference to construct the consensus sequence for low coverage regions. An annotation is added for the low coverage region in the consensus sequence produced (see below).

In addition to deciding how to handle low coverage regions, you can also decide how to handle conflicts or disagreement between the reads when building a consensus sequence in regions above the low coverage threshold:

- **Vote.** Whenever the reads disagree on the base at a given position, the vote resolution will let the majority of the reads decide which base is correct. In addition, you can specify to

let the voting use the base calling **quality scores** from the reads. This is done by simply adding all quality scores for each base and let the sum determine which one is correct. The base with the highest total quality scores will be chosen. If there are two bases that end up summing to the same total quality score for all reads at that location, A is preferred before C, C before G, and G before T. An annotation with the complete information that was used to resolve the conflict will be added.

- **Insert ambiguity codes.** When this option is selected, read conflicts are addressed by using an ambiguity code representing all read bases represented at the reference location. The problem with the voting option is that it will not be able to represent true biological heterozygous variation in the data. For a diploid genome, if two different alleles are present in an almost even number of reads, only one will be represented in the consensus sequence. With the option to insert ambiguity codes, this can be solved. However, if an ambiguity code would always be inserted if just one read had a different base, there would be an ambiguity code whenever there was a sequencing error. In high-coverage NGS data that would be a big problem, because sequencing errors would be abundant. To solve this problem, you can specify a **Noise threshold**. The default value for this is 0.1 which means that for a base to contribute to the ambiguity code, it must be in at least 10 % of the reads at a given position. The **Minimum nucleotide count** specifies the minimum number of reads that are required before a nucleotide is included. Nucleotides below this limit are considered noise. If no nucleotide passes the noise filters, the position is omitted from the consensus sequence.
- **Use quality score.** The "Use quality score" checkbox option is available for conflicts regardless of whether "Vote" or "Insert ambiguity codes" has been selected. The "Use quality score" checkbox option allows you to use the base calling **quality scores** from the reads. This is done by simply adding all the quality scores for each base and let the sum determine which bases to consider. In other words, if quality scores are used, we will sum the quality score (instead of amount of reads) for each base on each position before applying the noise filters and finally call the consensus symbol.

Click **Next** to set the output option as shown in figure 23.12).

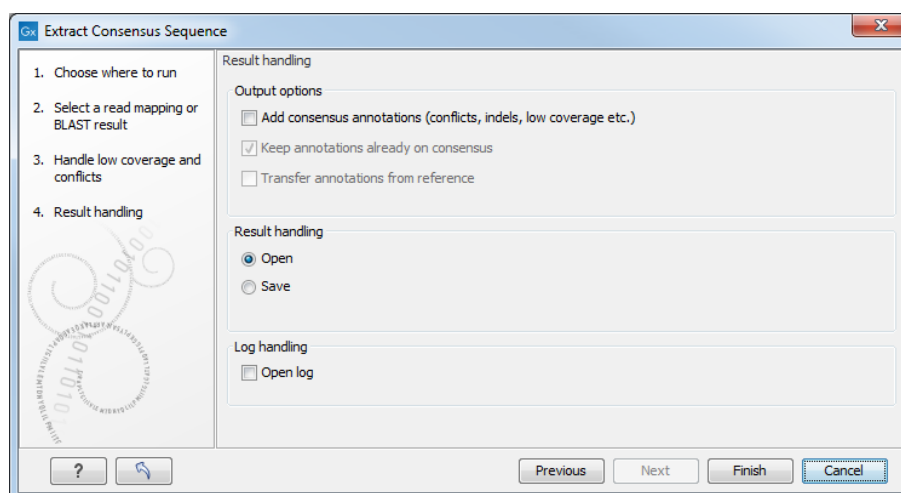


Figure 23.12: Choose to add annotations to the consensus sequence.

The annotations that can be added to the consensus sequence produced by this tool show both

conflicts that have been resolved and low coverage regions (unless you have chosen to split the consensus sequence). Please note that for large data sets, this can amount to a very high number of annotations, which will cause the tool to take longer to complete, and the result will take up much more disk space.

It is also possible to transfer existing annotations to the consensus sequence produced. Please note that since the consensus sequence produced may be broken up, the annotations will also be broken up, and you cannot expect them to have the same length as before. In some cases, gaps and low-coverage regions will lead to differences in the sequence coordinates between the input data and the new consensus sequence. The annotations copied will be placed in the region on the consensus that corresponds to the region on the input data, but the actual coordinates might have changed.

Copied/transferred annotations will contain the same qualifier text as the original. That is, the text is not updated. As an example, if the annotation contains 'translation' as qualifier text this translation will be copied to the new sequence and will thus reflect the translation of the original sequence, not the new sequence, which may differ.

The resulting consensus sequence (or sequences) will have quality scores assigned if quality scores were found in the reads used to call the consensus. For a given consensus symbol X we compute its quality score from the "column" in the read mapping. Let Y be the sum of all quality scores corresponding to the "column" below X , and let Z be the sum of all quality scores from that column that supported X ¹. Let $Q = Z - (Y - Z)$, then we will assign X the quality score of q where

$$q = \begin{cases} 64 & \text{if } Q > 64 \\ 0 & \text{if } Q < 0 \\ Q & \text{otherwise} \end{cases}$$

23.4 Local BLAST databases

BLAST databases on your local system can be made available for searches via your *CLC Main Workbench*, (section 23.4.1). To make adding databases even easier, you can download pre-formatted BLAST databases from the NCBI from within your *CLC Main Workbench*, (section 23.4.2). You can also easily create your own local blast databases from sequences within your *CLC Main Workbench*, (section 23.4.3).

23.4.1 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 23.5). All the files that comprise a given BLAST database must be included. This may be as few as three files, but can be more (figure 23.13).
- Add the location where your BLAST databases are stored using the BLAST database manager (see section 23.5) (figure 23.17).

¹By supporting a consensus symbol, we understand the following: when conflicts are resolved using voting, then only the reads having the symbol that is eventually called are said to support the consensus. When ambiguity codes are used instead, all reads contribute to the called consensus and thus $Y = Z$.

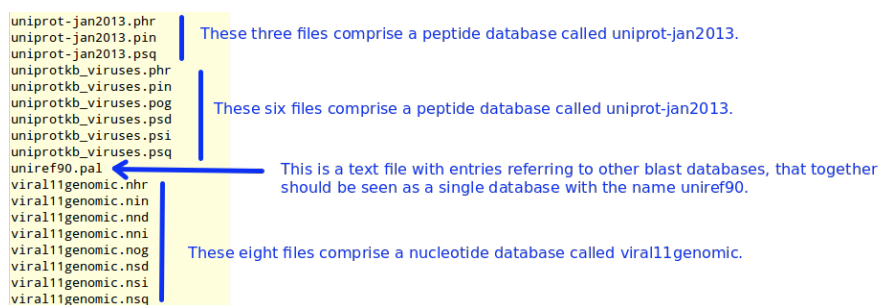


Figure 23.13: BLAST databases are made up of several files. The exact number varies and depends on the tool used to build the databases as well as how large the database is. Large databases will be split into the number of volumes and there will be several files per volume. If you have made your BLAST database, or downloaded BLAST database files, outside the Workbench, you will need to ensure that all the files associated with that BLAST database are available in a CLC Blast database location.

23.4.2 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.

To download a database, go to:

Toolbox | BLAST (📁) | Download BLAST Databases (🌐)

A window like the one in figure 23.14 pops up showing you the list of databases available for download.

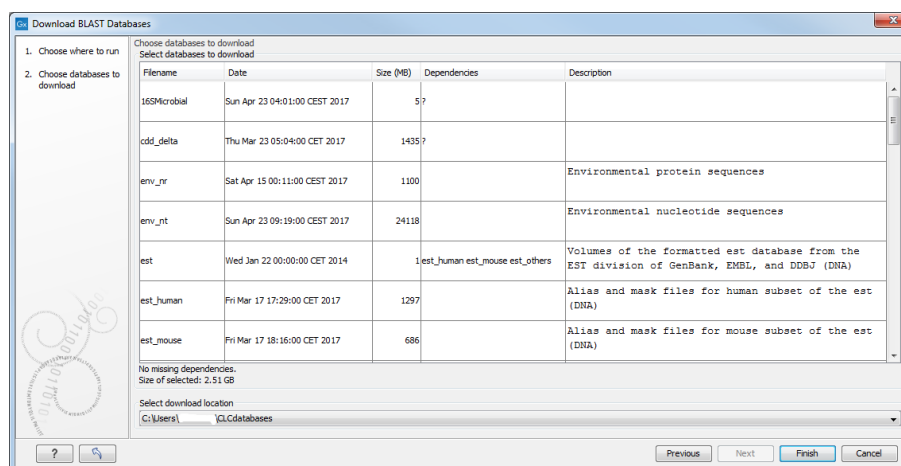


Figure 23.14: Choose from pre-formatted BLAST databases at the NCBI available for download.

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 23.5).

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 you 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.

23.4.3 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 23.1.2).

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 23.4.1.

To create a BLAST database, go to:

Toolbox | BLAST (📁) | Create BLAST Database (📁➕)

This opens the dialog seen in figure 23.15.

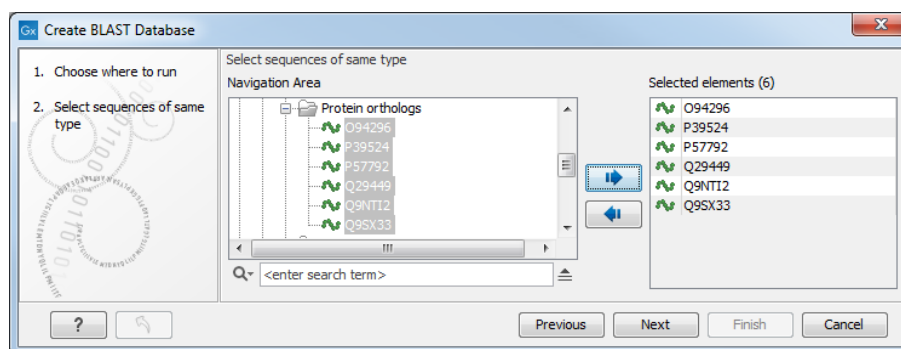


Figure 23.15: Add sequences for the BLAST database.

Select sequences or sequence lists you wish to include in your database and click **Next**.

In the next dialog, shown in figure 23.16, 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 23.5.

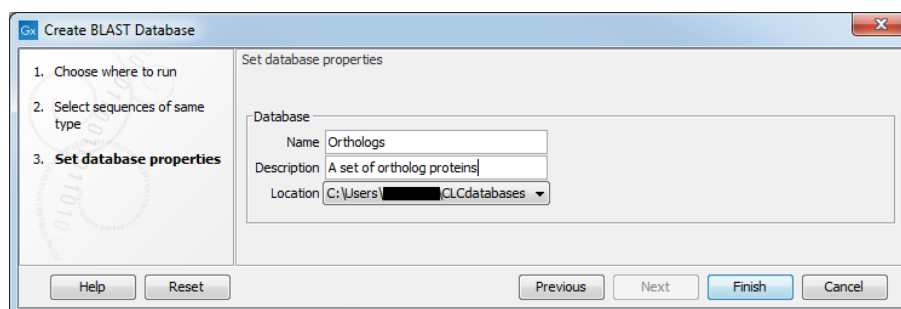


Figure 23.16: Providing a name and description for the database, and the location to save the files to.

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 23.5, and when running local BLAST (see section 23.1.2).

23.5 Manage BLAST databases

The BLAST databases available as targets for running local BLAST searches (see section 23.1.2) can be managed through the Manage BLAST Databases dialog (see figure 23.17):

Toolbox | BLAST (📁) | Manage BLAST Databases (🗑️)

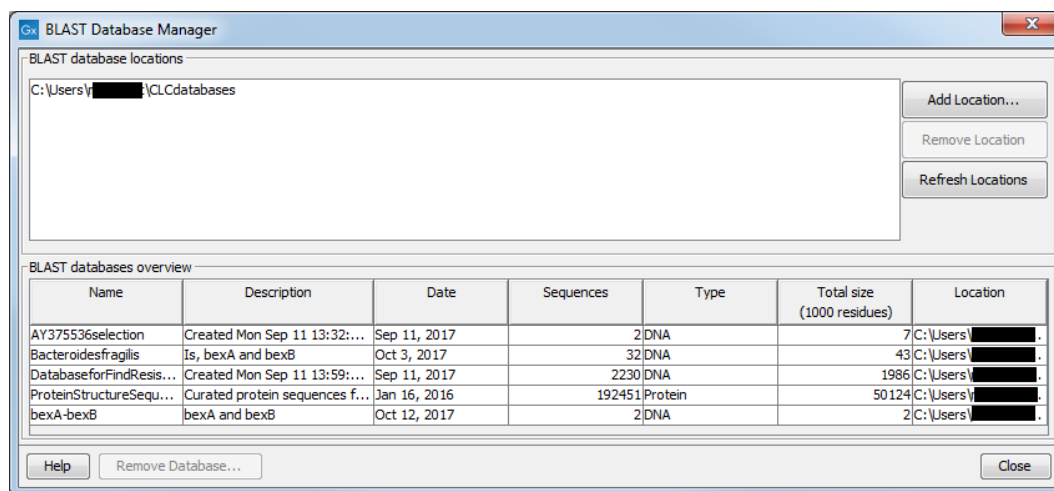


Figure 23.17: Overview of available BLAST databases.

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 23.4.3, 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.
- **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.

23.6 Bioinformatics explained: BLAST

BLAST (Basic Local Alignment Search Tool) has become the *defacto* standard in search and alignment tools [Altschul et al., 1990]. The BLAST algorithm is still actively being developed and is one of the most cited papers ever written in this field of biology. Many researchers use BLAST as an initial screening of their sequence data from the laboratory and to get an idea of what they are working on. BLAST is far from being basic as the name indicates; it is a highly advanced algorithm which has become very popular due to availability, speed, and accuracy. In short, a BLAST search identifies homologous sequences by searching one or more databases usually hosted by NCBI (<http://www.ncbi.nlm.nih.gov/>), on the query sequence of interest [McGinnis and Madden, 2004].

BLAST is an open source program and anyone can download and change the program code. This has also given rise to a number of BLAST derivatives; WU-BLAST is probably the most commonly used [Altschul and Gish, 1996].

BLAST is highly scalable and comes in a number of different computer platform configurations which makes usage on both small desktop computers and large computer clusters possible.

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.

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.

23.6.1 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 23.18 for an illustration of words in a protein sequence.

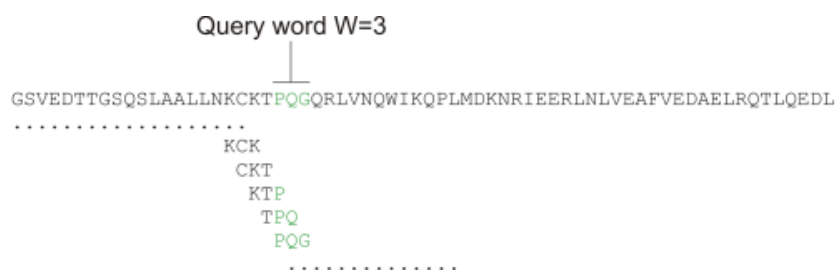


Figure 23.18: Generation of exact BLAST words with a word size of $W=3$.

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 23.18). 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 23.19). The default scoring matrix for blastp is BLOSUM62. The compilation of exact words and neighborhood words is then used to match against the database sequences.

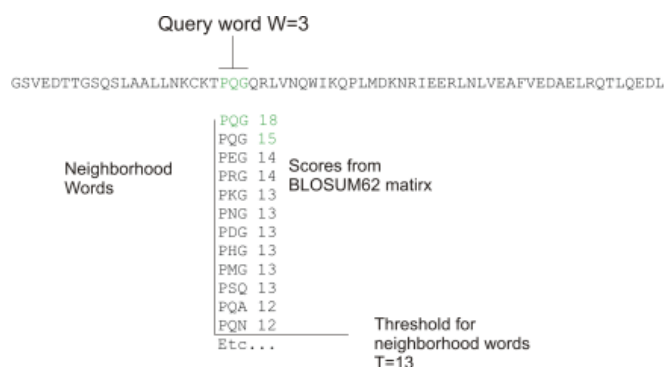


Figure 23.19: Neighborhood BLAST words based on the BLOSUM62 matrix. Only words where the threshold T exceeds 13 are included in the initial seeding.

After initial finding of words (seeding), the BLAST algorithm will extend the (only 3 residues long) alignment in both directions (see figure 23.20). 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.

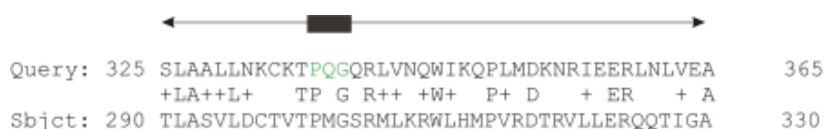


Figure 23.20: Blast aligning in both directions. The initial word match is marked green.

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 23.21.

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.

23.6.2 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,

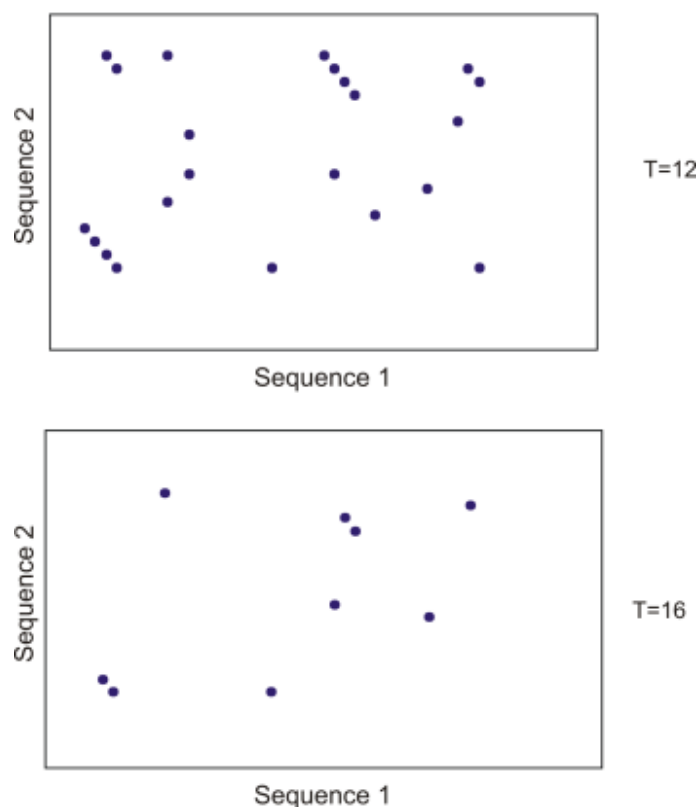


Figure 23.21: Each dot represents a word match. Increasing the threshold of T limits the search space significantly.

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.

23.6.3 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 'ffttflllsss', 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. The default scoring matrix for blastp is BLOSUM62.

23.6.4 Explanation of the BLAST output

The BLAST output comes in different flavors. On the NCBI web page the default output is html, and the following description will use the html output as example. Ordinary text and xml output for easy computational parsing is also available.

The default layout of the NCBI BLAST result is a graphical representation of the hits found, a table of sequence identifiers of the hits together with scoring information, and alignments of the query sequence and the hits.

The graphical output (shown in figure 23.22) gives a quick overview of the query sequence and the resulting hit sequences. The hits are colored according to the obtained alignment scores.

The table view (shown in figure 23.23) provides more detailed information on each hit and furthermore acts as a hyperlink to the corresponding sequence in GenBank.

In the alignment view one can manually inspect the individual alignments generated by the BLAST algorithm. This is particularly useful for detailed inspection of the sequence hit found (sbjct) and the corresponding alignment. In the alignment view, all scores are described for each alignment, and the start and stop positions for the query and hit sequence are listed. The strand and orientation for query sequence and hits are also found here.

In most cases, the table view of the results will be easier to interpret than tens of sequence alignments.

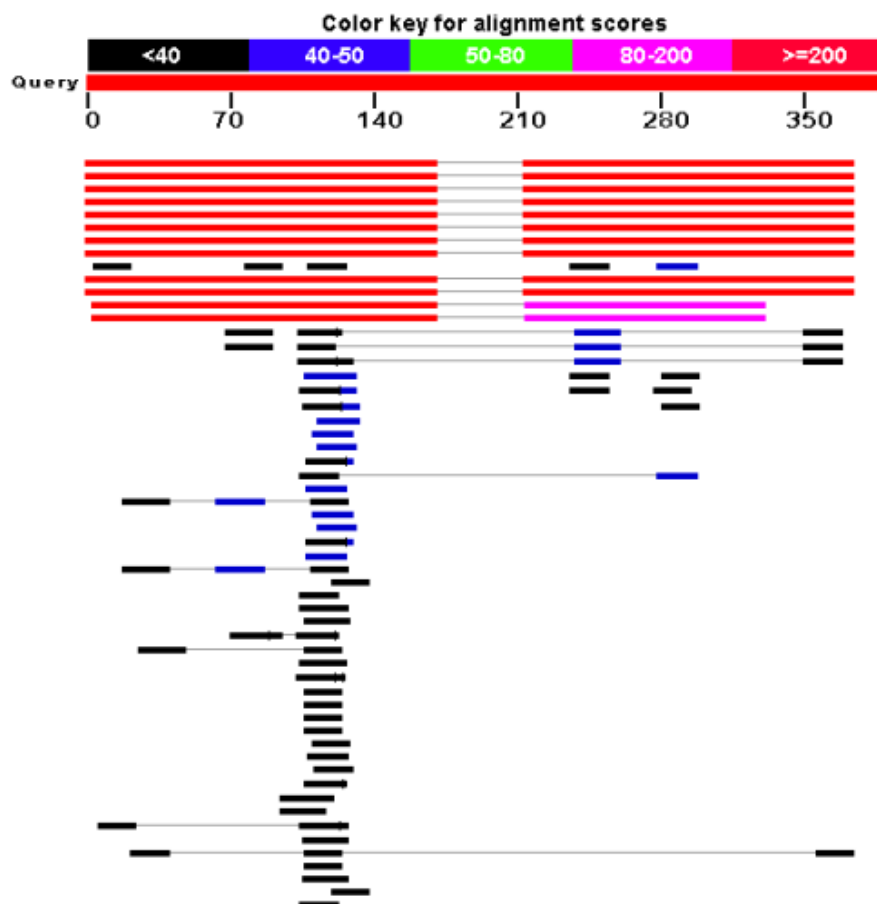


Figure 23.22: *BLAST graphical view.* A simple graphical overview of the hits found aligned to the query sequence. The alignments are color coded ranging from black to red as indicated in the color label at the top.

Sequences producing significant alignments:
(Click headers to sort columns)

Accession	Description	Max score	Total score	Query coverage	E value	Max ident	Links
Transcripts							
NM_174886.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173210.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173209.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173211.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173207.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_173208.1	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_170695.2	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_003244.2	Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),	339	563	85%	1e-90	100%	U E G M
NM_003246.2	Homo sapiens thrombospondin 1 (THBS1), mRNA	38.2	38.2	4%	7.2	100%	U E G M
NM_177965.2	Homo sapiens chromosome 8 open reading frame 37 (C8orf37),	38.2	38.2	4%	7.2	100%	U E G M
Genomic sequences [show first]							
NT_010859.14	Homo sapiens chromosome 18 genomic contig, reference assembly	339	602	85%	1e-90	100%	
NW_926940.1	Homo sapiens chromosome 18 genomic contig, alternate assembly	339	602	85%	1e-90	100%	
NT_011109.15	Homo sapiens chromosome 19 genomic contig, reference assembly	262	375	73%	3e-67	94%	
NW_927217.1	Homo sapiens chromosome 19 genomic contig, alternate assembly	262	375	73%	3e-67	94%	

Figure 23.23: *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.

23.6.5 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.


```

> ref|NM\_173209.1 UEGM Homo sapiens TGFB-induced factor (TALE family homeobox) (TGIF),
transcript variant 5, mRNA
Length=1382

Sort alignments for this subject sequence by:
E value Score Percent identity
Query start position Subject start position

Score = 339 bits (171), Expect = 1e-90
Identities = 171/171 (100%), Gaps = 0/171 (0%)
Strand=Plus/Plus

Query 1   ATTTGCACATGGGATTGCTAAACAGCTTCCTGTTACTGAGATGCTTCAATGGAATACA 60
          |||
Sbjct 993  ATTTGCACATGGGATTGCTAAACAGCTTCCTGTTACTGAGATGCTTCAATGGAATACA 1052

Query 61  GTCATCCAAGAACTATAAACTTAAAGCTACTGTAGAAACAAGGGTTTTCTTTTTTAAA 120
          |||
Sbjct 1053 GTCATCCAAGAACTATAAACTTAAAGCTACTGTAGAAACAAGGGTTTTCTTTTTTAAA 1112

Query 121 TGTTTCTTGGTAGATTATTCATAATGTGAGATGGTCCCAATATCATGTGA 171
          |||
Sbjct 1113 TGTTTCTTGGTAGATTATTCATAATGTGAGATGGTCCCAATATCATGTGA 1163

Score = 224 bits (113), Expect = 6e-56
Identities = 161/161 (100%), Gaps = 0/161 (0%)
Strand=Plus/Plus

Query 213 GACTGTGCAATACTTAGAGAACCCTATAGCATCTTCTCATTCCCATGTGGAACAGGATGCC 272
          |||
Sbjct 1205 GACTGTGCAATACTTAGAGAACCCTATAGCATCTTCTCATTCCCATGTGGAACAGGATGCC 1264

Query 273 CACATACTGTCTAATTAATAAAATTTTCCAtttttttCAAACAAGTATGAATCTAGTTGG 332
          |||
Sbjct 1265 CACATACTGTCTAATTAATAAAATTTTCCATTTTTTTCAAACAAGTATGAATCTAGTTGG 1324

Query 333 TTGATGCCttttttttCATGACATAATAAAGTATTTTCTTT 373
          |||
Sbjct 1325 TTGATGCCTTTTTTTTCATGACATAATAAAGTATTTTCTTT 1365

```

Figure 23.24: Alignment view of BLAST results. Individual alignments are represented together with BLAST scores and more.

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 23.25). It is also much easier to batch download a selection of hit sequences for further inspection.

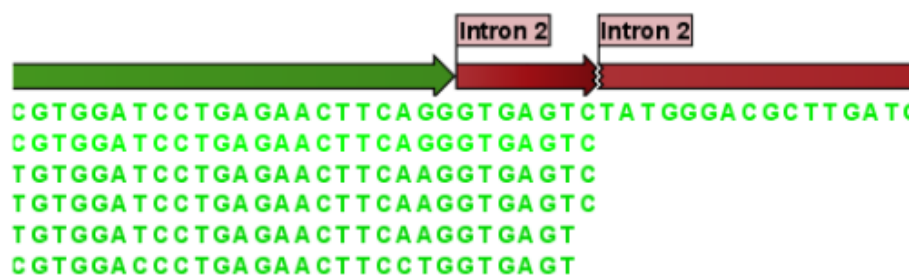


Figure 23.25: Snippet of alignment view of BLAST results. Individual alignments are represented directly in a graphical view. The top sequence is the query sequence and is shown with a selection of annotations.

23.6.6 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.

23.6.7 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/>

Chapter 24

Utility Tools

Contents

24.1 Batch Rename	547
24.2 Extract Annotations	552

24.1 Batch Rename

With the Batch Rename tool it is possible to rename your data in a batch fashion.

To run the batch rename tool:

Toolbox | Utilities (🔧) | Batch Rename (a|e)

This will open the dialog shown in figure 24.1 where you can select the input data.

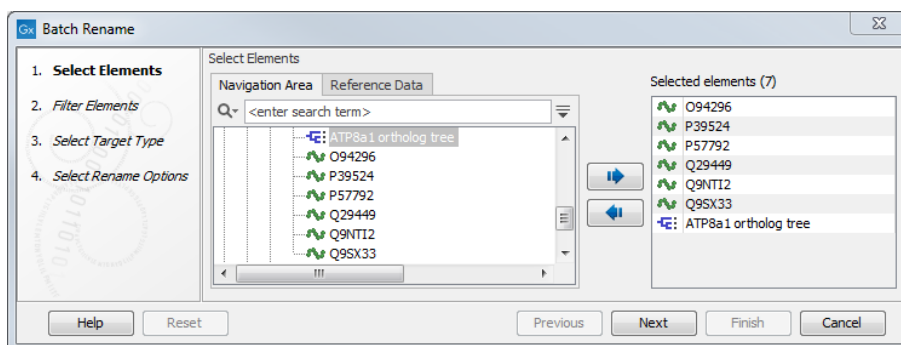


Figure 24.1: Selecting the data you want to rename.

Click on the button labeled **Next** to go to the next dialog (see figure 24.2).

Here, one can choose to include or exclude only some of the data previously selected to work on. For small numbers of data elements, this would not usually be necessary. However, if many data objects were selected at the previous step (to save time when choosing many data elements) you could use the include or exclude functionality at this point so that only certain data elements will be acted on by the batch rename tool.

The **Include** and **Exclude** filters take the text entered into the respective fields and search for matches in the names of the data elements selected in the first wizard step. Thus, you could

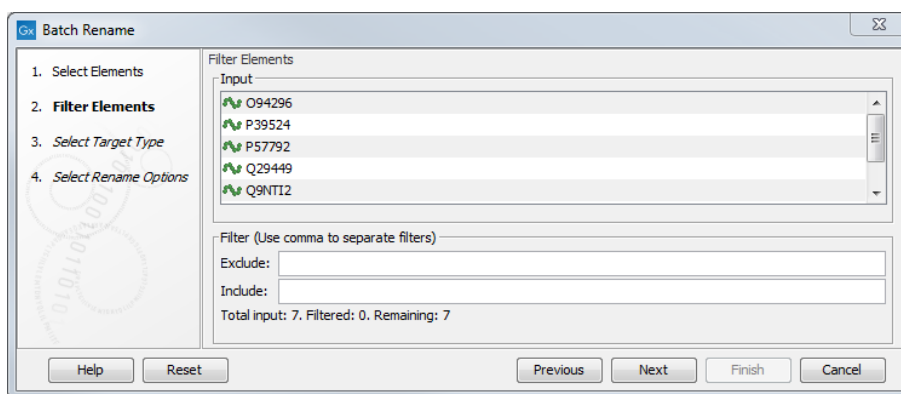


Figure 24.2: Select how to filter the input data.

enter the full names of particular data elements, or just partial names. Any elements where a match is found to the term or terms in the **Include** field will have the batch renaming applied to them. Any elements where a match is found to the term or terms in the **Exclude** field will not have the batch renaming applied to them.

For both filters, if you wish to filter on more than one term at the time, the individual terms must be separated with a comma - and without using a space after the comma. An example is shown in figure 24.3.

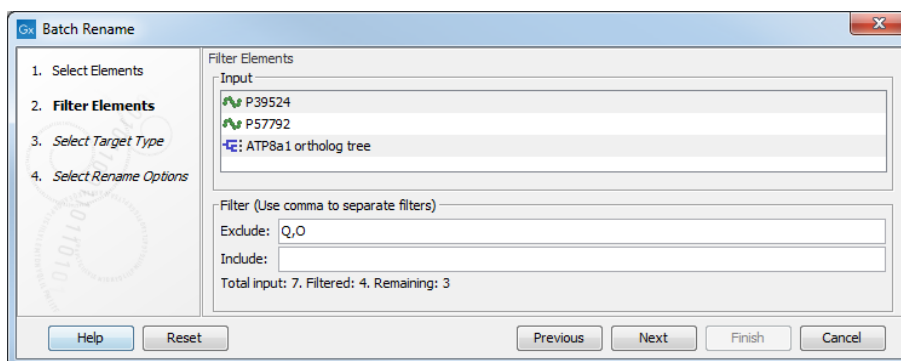


Figure 24.3: Items remaining after filtering away items with a Q or an O in their name.

In the "Select Target type" step, you can specify at which level you wish to perform the renaming. For a single sequence - as in the example shown in figure 24.4 - this is straightforward because it has just one name, and you would use the **Rename elements** option. But if you have a sequence list, for example, you could choose either to rename the list (using **Rename elements**) or the sequences in the list (using **Rename sequences in sequence lists**). The same goes for alignments (using **Rename sequences in alignments**) and read mappings (using **Rename reads in mappings**). For read mappings, there is also an option to **Rename reference sequence in mappings**.

Click on the button labeled **Next** to open the last dialog (see figure 24.5). For each text field, you can press Shift+F1 (Shift + Fn + F1 on Mac) to get a drop-down list of advanced placeholder options.

At this step you can select between three different renaming options.

- **Add text to name** This option will add text at the beginning or the end of the existing name, depending on which you choose. Pressing Shift + F1 (Shift + Fn + F1 on Mac) will

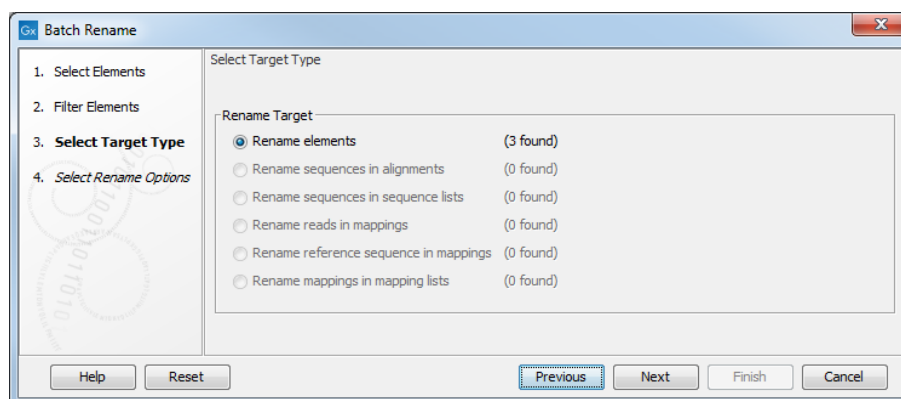


Figure 24.4: In this example, as we only have one category represented, the other target type options are disabled.

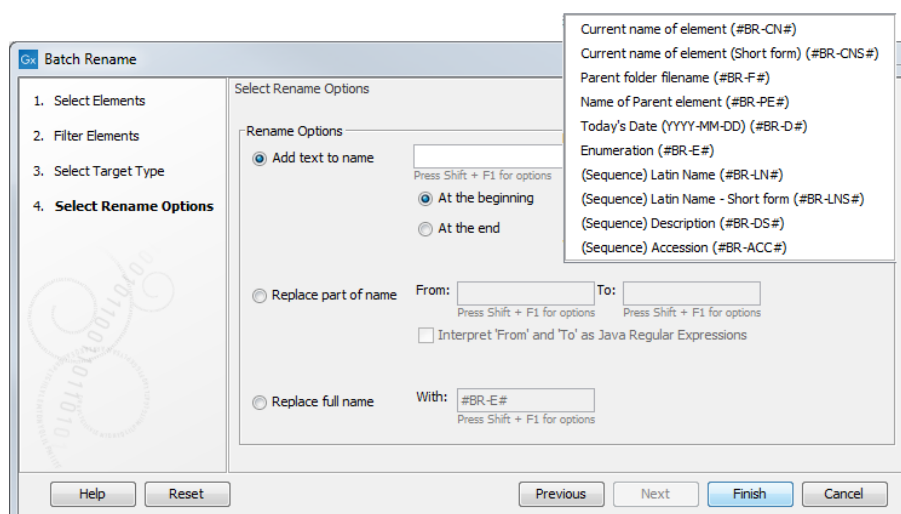


Figure 24.5: Rename options. At this step you can choose whether you wish to add text to the existing name, replace some of the name, or replace the entire name.

enable a list of different renaming options as shown in figure ???. The different options are presented as e.g. #BR-E#, which means "Batch Rename - Enumeration" = the current name is kept, and if "At the end" was selected, consecutive numbers will be added directly after the existing name (without introducing a space between the existing name and the new addition). Please note, that the numbering will follow the order of which the data were selected in the first dialog under "Select Elements".

- **Replace part of name** This is an advanced function that allows to replace e.g. data with completely different names in one go with a new name. This is shown in figure 24.6.

The option to replace part of a name is based on regular expressions. Regular expressions allows you to describe text in a flexible manner. For more details, please see: <http://docs.oracle.com/javase/tutorial/essential/regex/>.

By clicking in either the From: or To: box and pressing the Shift and F1 keys at the same time (Shift + Fn + F1 on Mac), you will see a drop down list of renaming possibilities. The options listed for the From: field are some of the most commonly used regular expressions. Other standard regular expressions are also admissible in this field. **Note!** Please ensure that if you choose any of these options, or other regular expressions, that you check in the box labeled "Interpret From as Java Regular Expressions". If you do not check this box,

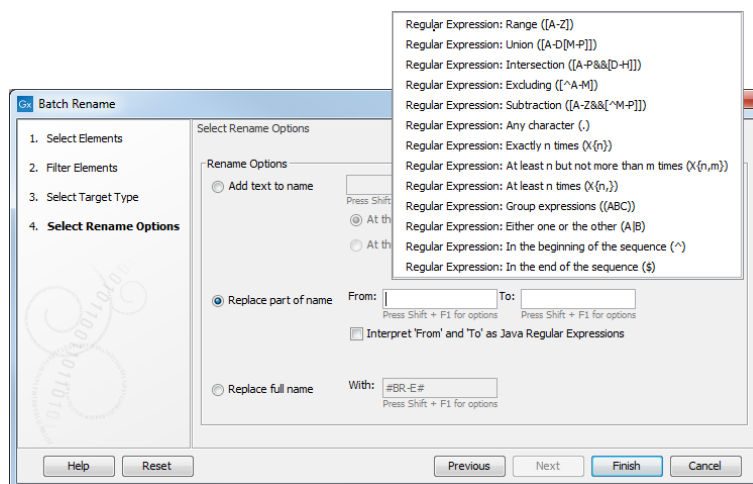


Figure 24.6: Advanced options for rename of only part of a name.

then the characters you enter in the From box will be interpreted literally. For example, a full stop or period (.) is interpreted as that character (.) when this box is not checked, but is interpreted as meaning any single character when the box is checked.

An example: if you enter "From:" "Regular Expression: Range ([A-Z])" "To:" "Enumeration (#BR-E#)", titles containing any (capital) letters will be renamed to consecutive numbers. A more advanced example where the sequences shown in the images earlier in this manual are used is shown in figure 24.7. The sequence names contain both a capital letter, small letters and a number. In the first, the number is kept and a date is added in front of the number.

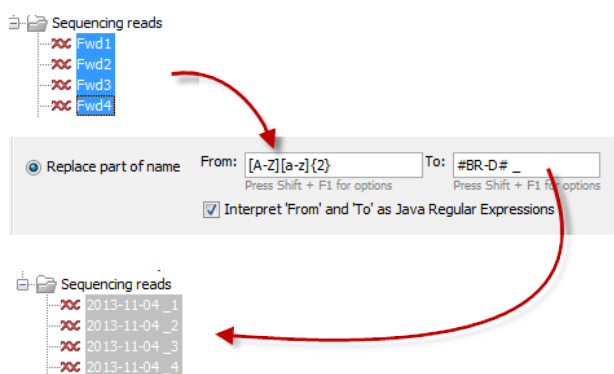


Figure 24.7: The letters "Fwd" are removed from the primer name and replaced with "Today's date". In this case we have also inserted a space and a "_" between the date and the number, which was the only thing that was left from the original primer name. Please note that in order to be able to remove both of the small letters in the primer name, you have to add 2, which indicates that [a-z] should be performed twice.

As this can be a bit difficult to grasp, we will look at three more examples. We will use the name **1N R1_0001** in the two first examples:

1. First we want to **keep only the first 4 non-whitespace characters of the name(s)**. To do this write the following in the "From" and "To" fields shown in figure 24.7:

From: (^\\S{4}).*

Nomenclature: ^: start of the line, \\S: non-whitespace characters,

{4}: 4 characters, .*: everything after the pattern

To: \$1

Nomenclature: \$1: the first group in the "From" field

The result of this is that *1NR1* is kept, whereas the space between "N" and "R1" and *_0001* have been discarded.

2. If we would like to **keep only the last 4 characters of the name(s)**:

From: (.*)(.{4}\$)

Nomenclature: (.*): the first group, (.{4}\$): any 4 character at the end of the line as the second group

To: \$2

Nomenclature: \$2: the second group in the "From" field

The result of this is that *0001* is kept, whereas *1N R1_* has been discarded.

3. Now we would like to **replace the first letter followed by 9 numbers in the name "p140101034_1R_AMR" with the parent folder name, which in this case is "AmericanSamples"**.

From: ([A-Z]\d{9})(.*)

Nomenclature: [A-Z]: any character

(as long as they are part of [A-Z][a-z], the CLC software do not differentiate between upper/lower case),

\d{9}: any 9 digit numbers, (.*): the second group of the name (anything after the "[A-Z]\d{9}" pattern).

To: #BR-F#\$2

Nomenclature: \$2: the second group in the "From" field.

The result of this is that we have replaced "p140101034" with "AmericanSamples" and as a result have changed the name from *p140101034_1R_AMR* to *AmericanSamples_1R_AMR*.

4. If we want to extract and use the text "sample-code" for the new name from the following "1234_sample-code_5678" with Java Regular Expressions:

From: (^[\^_]+)_([\^_]+)_(.*)

Nomenclature: ^[\^_]+: Starting from the beginning and match anything before the first underscore,

[\^_]+: will match anything and stop before the second underscore.

After the second underscore, the match will include the rest of the name.

To: \$2

Nomenclature: \$2: includes the match from the second group, which is flanked between the first and the second underscore symbols of the name.

- **Replace full name** Allows replacement of the entire name with the name that is either typed directly into the text field, or with options that can be selected when pressing Shift + F1 (Shift + Fn + F1 on Mac). Figure 24.8 shows an example where a combination of "Shift +F1" (Shift + Fn + F1 on Mac) options (#BR-D# and#BR-E#) are used together with user-defined text (RNA-Seq).

Click **Finish** to start renaming. Please note that the **rename cannot be undone** and that it does not show up in the **History** (📄).

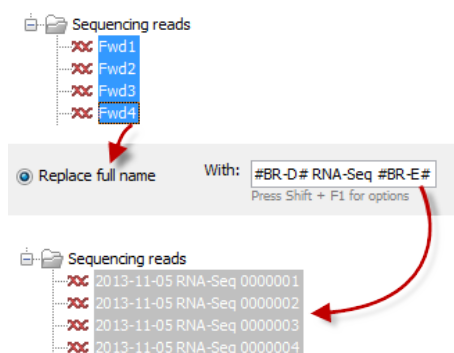


Figure 24.8: The entire name is removed from the primer names and is replaced with "Today's date" (#BR-D#), the userdefined text: RNA-Seq, and the addition of consecutive numbers (#BR-E#). In this case we have inserted a space between the date, the user-defined text and the added number. If commas were inserted instead, the commas would be part of the new name as everything that is typed into the text field will be used in the new name when renaming the entire name.

24.2 Extract Annotations

The **Extract annotations** tool makes it very easy to extract parts of a sequence (or several sequences) based on its annotations. In just a few steps, it becomes possible to:

- extract all tRNA genes from a genome.
- automatically add flanking regions to the annotated sequences.
- search for specific words in all available annotations.
- extract nucleotide sequences of differentially expressed genes or transcripts when using RNA-seq statistical comparisons as input.

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, go to:

Toolbox | Utility Tools | Extract Annotations (👉)

This opens the dialog shown in figure 24.9 that allows specification of which sequence to extract annotations from.

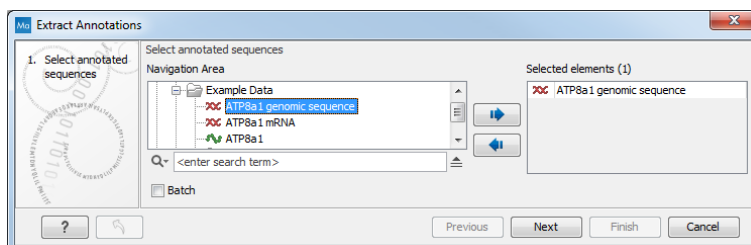


Figure 24.9: Select one or more sequences to extract annotations from.

Click **Next**. At the top of the dialog shown in figure 24.10 you can specify which annotations to use:

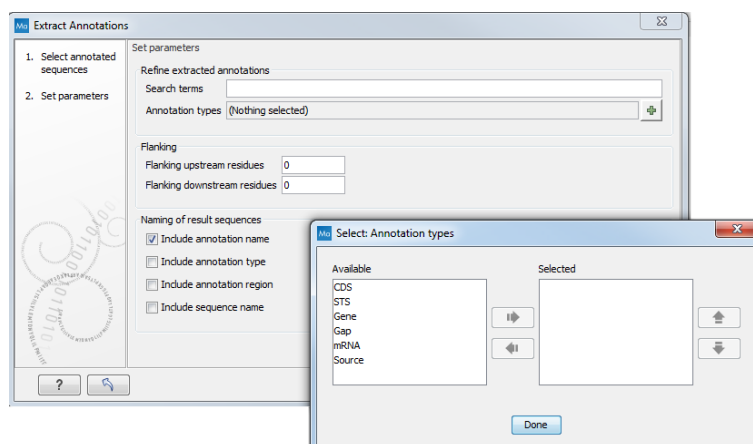


Figure 24.10: Adjusting parameters for extract annotations.

- **Search terms.** 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. For example, if you have a gene 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: "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 **Finish** to start the tool.

Part IV

Appendix

Appendix A

Graph preferences

This section explains the view settings of graphs. The **Graph preferences** at the top of the **Side Panel** includes the following settings:

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
 - Outside
 - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
 - None
 - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **X-axis at zero.** This will draw the x axis at $y = 0$. Note that the axis range will not be changed.
- **Y-axis at zero.** This will draw the y axis at $x = 0$. Note that the axis range will not be changed.
- **Show as histogram.** For some data-series it is possible to see the graph as a histogram rather than a line plot.

The **Lines and plots** below contains the following settings:

- **Dot type**

- None
- Cross
- Plus
- Square
- Diamond
- Circle
- Triangle
- Reverse triangle
- Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

- **Line width**

- Thin
- Medium
- Wide

- **Line type**

- None
- Line
- Long dash
- Short dash

- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

For graphs with multiple data series, you can select which curve the dot and line preferences should apply to. This setting is at the top of the **Side Panel** group.

Note that the graph title and the axes titles can be edited simply by clicking with the mouse. These changes will be saved when you **Save** (☒) the graph - whereas the changes in the **Side Panel** need to be saved explicitly (see section 4.6).

Appendix B

BLAST databases

Several databases are available at NCBI, which can be selected to narrow down the possible BLAST hits.

B.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..

B.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 Sagarssso Sea project. This does overlap with nucleotide nr.

B.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: https://web.archive.org/web/20120409025527/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.resources.qiagenbioinformatics.com/wbsettings/NCBI_BlastNucleotideDatabases.zip
- Protein databases: http://www.resources.qiagenbioinformatics.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[clcddefault] = 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 https://web.archive.org/web/20120409025527/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 C

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 D

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.resources.qiagenbioinformatics.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.

Note that in macOS, the extension file "link_emboss_e_custom" will have a ".txt" extension in its filename and metadata that needs to be removed. Right click the file name, choose "Get info" and remove ".txt" from the "Name & extension" field.

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 E

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.resources.qiagenbioinformatics.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 E.1).

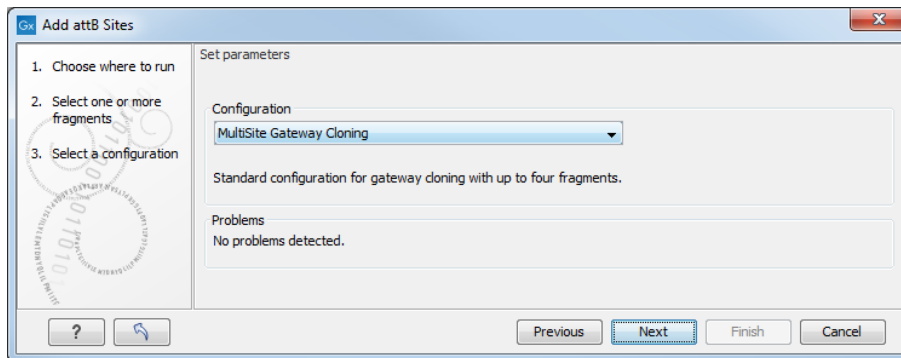


Figure E.1: Selecting between different gateway cloning configurations.

Appendix F

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 Isoleucine
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
Z	Glx	Glutamic acid or Glutamine
X	Xaa	Any amino acid

Appendix G

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 H

Formats for import and export

H.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.

H.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
Clone manager	.cm5	X		Clone manager sequence format
DNAstrider	.str/.strider	X	X	
DS Gene	.bsml	X		
EMBL	.emb/.embl	X	X	Rich information incl. annotations (nucs only)
FASTA	.fsa/.fasta	X	X	Simple format, name & description
GenBank	.gbk/.gb/.gp/.gbk	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	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 1.5).

When exporting in fasta format, it is possible to remove sequence ends covered by annotations of type "Trim" (read more in section 18.2).

H.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

H.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	

H.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	

H.1.5 Expression data formats

Read about technical details of these data formats in section I.

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
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
Table CSV	.csv		X	Samples and experiments
Tab delimited	.txt		X	Samples and experiments

H.1.6 Other formats

File type	Suffix	Import	Export	Description
CLC	.clc	X	X	Rich format including all information
PDB	.pdb	X		3D structure
RNA structures	.ct, .col, .rnaml/.xml	x		Secondary structure for RNA

H.1.7 Table and text formats

File type	Suffix	Import	Export	Description
Excel	.xls/.xlsx	X	X	All tables and reports
Table CSV	.csv	X	X	All tables
Tab delimited	.txt		X	All tables
Text	.txt	X	X	All data in a textual format
CLC	.clc	X	X	Rich format including all information
HTML	.html		X	All tables
PDF	.pdf		X	Export reports in Portable Document Format

Please see table [H.1.5 Expression data formats](#) for special cases of table imports.

H.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.

H.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 [6.3](#) 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 I

Gene expression annotation files and 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 [I.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 section [I.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 [I.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.

I.1 GEO (Gene Expression Omnibus)

The GEO (Gene Expression Omnibus) sample and series formats are supported. Figure [I.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,

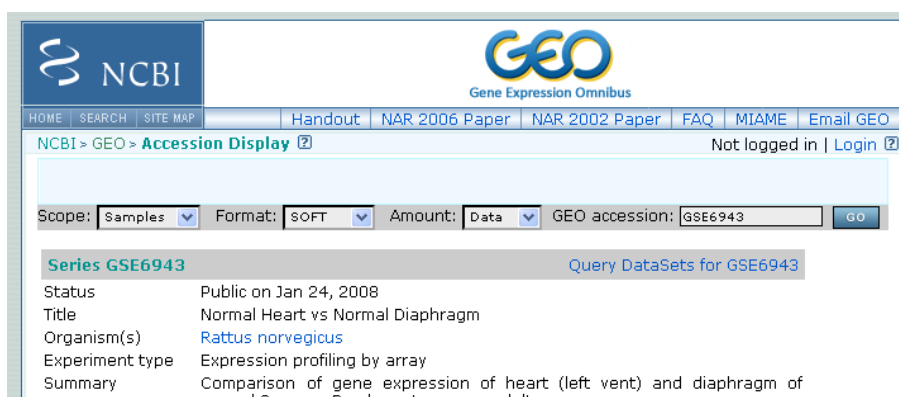


Figure I.1: Selecting Samples, SOFT and Data before clicking go will give you the format supported by the **CLC Main Workbench**.

lines are the column contents of the sample.

Note that GEO sample importer will also work for concatenated GEO sample files – allowing multiple samples to be imported in one go. Download a sample file containing concatenated sample files here:

<http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFilesConcatenated.txt>

Below you can find examples of the formatting of the GEO formats.

I.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.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimple.txt>

I.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.resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresent.txt>

I.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.resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresent.txt>

I.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.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequence.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.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequence.txt>

I.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.resources.qiagenbioinformatics.com/madata/GEOSeriesFile.txt>

I.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 probe-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 probe 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 1.5; 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.

If multiple probes are present for the same gene, further processing may be required to merge them into a single gene-level expression.

1.2.1 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.resources.qiagenbioinformatics.com/madata/AffymetrixCHPandPSI.zip>

1.2.2 Affymetrix metrics 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.resources.qiagenbioinformatics.com/madata/AffymetrixMetrics.txt>

I.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 22.1.3.

Download a small example annotation file here which includes header information:

<http://www.resources.qiagenbioinformatics.com/madata/AffymetrixNetAffxAnnotation.csv>

I.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.

I.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.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipCompact.txt>

I.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.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipExtended.txt>

I.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.resources.qiagenbioinformatics.com/madata/IlluminaBeadStudioWithAnnotation.txt>

I.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.resources.qiagenbioinformatics.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.

I.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 22.1.3.

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.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipAnnotation.txt>

I.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*. They can be used to annotate experiments as shown in section 22.1.3. They can also be used with the Gene Set Test and Create Expression Browser tools.

Import GO annotation file using the Standard Import tool. For GO annotation files in GAF format, use the option "Force import as type: Gene Ontology Annotation file" from the drop down menu at the bottom of the Standard Import dialog.

See the complete list of available files, including download links, at <http://www.geneontology.org/GO.current.annotations.shtml>.

I.5 Generic expression and annotation data file formats

If you have your expression or annotation data in 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.

I.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.resources.qiagenbioinformatics.com/madata/CustomExpressionData.txt>

I.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 identifiers 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.resources.qiagenbioinformatics.com/madata/SimpleCustomAnnotation.csv>

<http://www.resources.qiagenbioinformatics.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, ProbelD, 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
Ensembl 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	Transcript assignments
mma_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 J

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.

When creating a custom table, it is not necessary to fill in all fields as only the codon information (e.g. 'GCG' in the example below) and the counts (e.g. 47869.00) are used when doing reverse translation:

```
Name: Rattus norvegicus GeneticCode: 1 Ala GCG 47869.00 6.86 0.10 Ala GCA 109203.00  
15.64 0.23 ....
```

In particular, the amino acid type is not used: in order to use an alternative genetic code, it must be specified in the 'GeneticCode' line instead.

Restart the Workbench to have the changes take effect.

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., Studholme, D. J., Yeats, C., and Eddy, S. R. (2004). The Pfam protein families database. *Nucleic Acids Res.*, 32(Database issue):D138–D141.
- [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.
- [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.
- [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., Wüning, 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.
- [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.
- [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.
- [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.
- [Michener and Sokal, 1957] Michener, C. and Sokal, R. (1957). A quantitative approach to a problem in classification. *Evolution*, 11:130–162.
- [Mortazavi et al., 2008] Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by rna-seq. *Nat Methods*, 5(7):621–628.
- [Mukherjee and Zhang, 2009] Mukherjee, S. and Zhang, Y. (2009). MM-align: A quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming. *Nucleic Acids Res.*, 37.
- [Purvis, 1995] Purvis, A. (1995). A composite estimate of primate phylogeny. *Philos Trans R Soc Lond B Biol Sci*, 348(1326):405–421.
- [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.
- [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.
- [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.
- [Xu and Zhang, 2010] Xu, J. and Zhang, Y. (2010). How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics*, 26(7):889–95.
- [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.
- [Zhang and Skolnick, 2004] Zhang, Y. and Skolnick, J. (2004). Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–10.
- [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 Mathematical 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.