This software is for research purposes only.
# Contents

## I Introduction

1 Introduction to CLC Main Workbench 12

1.1 Contact information and citation 13
1.2 Download and installation 14
1.3 System requirements 18
1.4 Workbench Licenses 19
1.5 Plugins 34
1.6 Network configuration 36
1.7 History of the CLC Workbenches 37

## II Core Functionalities

2 User interface 40

2.1 View Area 41
2.2 Zoom functionality in the View Area 49
2.3 Toolbox and Favorites tabs 51
2.4 Processes tab and Status bar 54
2.5 History and Element Info views 55
2.6 Workspace 56
2.7 List of shortcuts 58

3 Data management and search 61

3.1 Navigation Area 61
3.2 Working with tables 71
3.3 Customized attributes on data locations 75
CONTENTS

3.4 Searching for data in CLC Locations .................................................. 79

4 User preferences and settings ................................................................. 85
   4.1 General preferences ................................................................. 86
   4.2 View preferences ................................................................. 87
   4.3 Data preferences ................................................................. 90
   4.4 Advanced preferences ......................................................... 90
   4.5 Export/import of preferences ................................................. 92
   4.6 Side Panel view settings ..................................................... 93

5 Printing ............................................................................................... 96
   5.1 Selecting which part of the view to print ........................................ 97
   5.2 Page setup ............................................................................. 98
   5.3 Print preview ......................................................................... 100

6 Connections to other systems ............................................................... 101
   6.1 CLC Server connection ........................................................... 101
   6.2 AWS Connections .................................................................. 103

7 Import/export of data and graphics ......................................................... 106
   7.1 Standard import .................................................................... 106
   7.2 Data export ........................................................................... 108
   7.3 Export graphics to files ......................................................... 123
   7.4 Export graph data points to a file ........................................... 128
   7.5 CLC Server data import and export ....................................... 129
   7.6 Copy/paste view output ....................................................... 130

8 Data download .................................................................................. 131
   8.1 Search for Sequences at NCBI .................................................. 131
   8.2 Search for PDB Structures at NCBI ......................................... 134
   8.3 Search for Sequences in UniProt (Swiss-Prot/TrEMBL) ............... 137
   8.4 Sequence web info .............................................................. 140

9 Running tools, handling results and batching ........................................ 143
   9.1 Running tools ....................................................................... 143
CONTENTS

14.1 Create an alignment .................................................. 290
14.2 View alignments ...................................................... 295
14.3 Edit alignments ....................................................... 299
14.4 Join alignments ...................................................... 304
14.5 Pairwise comparison .................................................. 305

15 Phylogenetic trees .................................................. 310
15.1 K-mer Based Tree Construction .................................. 312
15.2 Create tree ............................................................. 313
15.3 Model Testing .......................................................... 314
15.4 Maximum Likelihood Phylogeny .................................. 316
15.5 Tree Settings ............................................................ 324
15.6 Metadata and phylogenetic trees .................................. 333

16 General sequence analyses .......................................... 339
16.1 Annotate with GFF/GTF/GVF file ................................. 339
16.2 Extract sequences ..................................................... 342
16.3 Shuffle sequence ..................................................... 343
16.4 Dot plots ............................................................... 344
16.5 Local complexity plot ............................................... 353
16.6 Sequence statistics .................................................. 353
16.7 Join Sequences ........................................................ 358
16.8 Pattern discovery ..................................................... 359
16.9 Motif Search ........................................................... 361
16.10 Create motif list ...................................................... 366

17 Nucleotide analyses .................................................. 368
17.1 Convert DNA to RNA ................................................ 368
17.2 Convert RNA to DNA ................................................ 369
17.3 Reverse complements of sequences ............................... 369
17.4 Translation of DNA or RNA to protein ............................ 370
17.5 Find open reading frames ......................................... 372

18 Protein analyses ...................................................... 375
## CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>20.8</td>
<td>Sequencing primers</td>
<td>451</td>
</tr>
<tr>
<td>20.9</td>
<td>Alignment-based primer and probe design</td>
<td>452</td>
</tr>
<tr>
<td>20.10</td>
<td>Analyze primer properties</td>
<td>457</td>
</tr>
<tr>
<td>20.11</td>
<td>Find binding sites and create fragments</td>
<td>458</td>
</tr>
<tr>
<td>20.12</td>
<td>Order primers</td>
<td>462</td>
</tr>
<tr>
<td>21.1</td>
<td>restriction site analyses</td>
<td>464</td>
</tr>
<tr>
<td>21.2</td>
<td>Restriction enzyme lists</td>
<td>473</td>
</tr>
<tr>
<td>21.3</td>
<td>Restriction Based Cloning</td>
<td>475</td>
</tr>
<tr>
<td>21.4</td>
<td>Homology Based Cloning</td>
<td>483</td>
</tr>
<tr>
<td>21.5</td>
<td>Gateway cloning</td>
<td>492</td>
</tr>
<tr>
<td>21.6</td>
<td>Gel electrophoresis</td>
<td>498</td>
</tr>
<tr>
<td>22.1</td>
<td>RNA secondary structure prediction</td>
<td>502</td>
</tr>
<tr>
<td>22.2</td>
<td>View and edit secondary structures</td>
<td>508</td>
</tr>
<tr>
<td>22.3</td>
<td>Evaluate structure hypothesis</td>
<td>516</td>
</tr>
<tr>
<td>22.4</td>
<td>Structure scanning plot</td>
<td>518</td>
</tr>
<tr>
<td>22.5</td>
<td>Bioinformatics explained: RNA structure prediction by minimum free energy minimization</td>
<td>520</td>
</tr>
<tr>
<td>23.1</td>
<td>Experimental design</td>
<td>527</td>
</tr>
<tr>
<td>23.2</td>
<td>Transformation and normalization</td>
<td>539</td>
</tr>
<tr>
<td>23.3</td>
<td>Quality control</td>
<td>543</td>
</tr>
<tr>
<td>23.4</td>
<td>Feature clustering</td>
<td>555</td>
</tr>
<tr>
<td>23.5</td>
<td>Statistical analysis - identifying differential expression</td>
<td>563</td>
</tr>
<tr>
<td>23.6</td>
<td>Annotation tests</td>
<td>570</td>
</tr>
<tr>
<td>23.7</td>
<td>General plots</td>
<td>576</td>
</tr>
<tr>
<td>24.1</td>
<td>Running BLAST searches</td>
<td>583</td>
</tr>
<tr>
<td>24.2</td>
<td>Output from BLAST searches</td>
<td>590</td>
</tr>
</tbody>
</table>
Part I

Introduction
Chapter 1

Introduction to CLC Main Workbench

Welcome to CLC Main Workbench 23.0.5 — a software package supporting your daily bioinformatics work.

CLC Main Workbench 23.0.5 is for research purposes only.

The CLC Main Workbench provides an easy-to-use graphical interface for running bioinformatics analyses. Tools can be run individually, or chained together in a workflow, making running
complex analyses simple and efficient. The functionality of the CLC Main Workbench can also be extended using plugins. The built-in Plugin Manager provides an up-to-date listing. A list is also available on our plugin webpage: https://digitalinsights.qiagen.com/products-overview/plugins/.

Supporting documentation and links for the CLC Main Workbench can be found under the Help menu in the top toolbar. Of particular note when getting started:

- The built-in Workbench user manual can be opened by choosing the Help option or by clicking on the F1 key.
- Manuals for installed plugins can be accessed under the Plugin Help option.
- The Online Tutorials option opens our tutorials webpage in a browser. Tutorials offer hands-on examples of how to use features of the CLC Main Workbench. Alternatively, click on the following link to visit that webpage: https://digitalinsights.qiagen.com/support/tutorials/.

Watch product specialists demonstrates our software in the videos offered via our Online presentations area: http://tv.qiagenbioinformatics.com/.

The latest version of this user manual can be found in pdf and html formats at https://digitalinsights.qiagen.com/technical-support/manuals/

The 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 https://digitalinsights.qiagen.com/technical-support/latest-improvements/.

### 1.1 Contact information and citation

CLC Main Workbench is developed by:

QIAGEN Aarhus
Silkeborgvej 2
Prismet
8000 Aarhus C
Denmark

https://digitalinsights.qiagen.com/

Email: ts-bioinformatics@qiagen.com

The QIAGEN Aarhus team is continuously improving CLC Main Workbench with your 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 CLC Main Workbench works, and the license you are using is covered by our Maintenance, Upgrades and Support (MUS) program (https://digitalinsights.qiagen.com/...
technical-support/maintenance-and-support/), please contact us by clicking on the Support button at the right hand side of the top toolbar (figure 1.1) or by choosing Contact Support under the Help menu.

Figure 1.1: Contact our Support team by clicking on the button at the right hand side of the top Toolbar

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 the license(s) being used by a CLC Workbench and any installed modules can be found by opening the License Manager:

- Help | License Manager...

Information about MUS cover on particular licenses is provided in your myCLC account: https://secure.clcbio.com/myclc/login.

How to cite us To cite a CLC Workbench or Server product, use the name of the product, the version number. For example QIAGEN CLC Main Workbench 21.0 or QIAGEN CLC Genomics Workbench 21.0. If a location is required by the publisher of the publication, use (QIAGEN, Aarhus, Denmark). Our website is https://digitalinsights.qiagen.com/.

Further details about citing QIAGEN Digital Insights software can be found in our FAQ at https://qiagen.secure.force.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA41000000L63hCAC

1.2 Download and installation

Software installers for Windows, macOS and Linux systems for the current CLC Main Workbench release can be downloaded from https://digitalinsights.qiagen.com/downloads/product-downloads/.

More ways to get installers is described in the Frequently Asked Question: "Where can I get installer files for the software?": https://qiagen.secure.force.com/KnowledgeBase/
To check for available updates from within the software, go to the menu option: Help | Check for Updates... (🪐).

General information about running software installers, including differences between upgrading to a new minor version compared to upgrading to a new major version, are covered in section 1.2.1. Detailed instructions for running the software installer in interactive mode on each supported operating system then follows.


### 1.2.1 General information about installing and upgrading Workbenches

When a Workbench installer is run, it performs the following tasks:

1. **Extract and copy files to the installation directory** The Workbench software is installed into a directory. It is self contained. The suggested folder name to install into reflects the software name and the major version line. For example, for a CLC Genomics Workbench with major version 23, the default installation location offered on each platform would be:

   **Windows** C:\Program files\CLC Genomics Workbench 23  
   **macOS** /Applications/CLC Genomics Workbench 23  
   **Linux** /opt/CLCGenomicsWorkbench 23

To install the software into central locations, like those listed above, generally requires administrator rights. Administrator rights will also be needed to install licenses and plugins for installations in central locations. The software can be installed to another location, if desired. When only a single person will use the software, this can be useful. Installing to an area they have permission to write to means that licenses and plugins can then be installed without needing administrator rights.

**General recommendations for installation locations**

- **For minor updates**, you will be asked whether you wish to:
  - **Update the existing installation** Generally recommended for minor updates. New files will be installed into the same directory as the existing installation. Licensing information and installed plugins remain in place from the installation already present.  
  OR  
  - **Install to a different directory**. Configuration will be needed after installation. E.g. licensing needs to be configured, any desired plugins will need to be installed, etc.

- **For major updates**. The suggested installation directory will reflect the new major version number of the software. Please do not install a new major version into the same folder as an existing, older version of the Workbench. Configuration will be needed after installation. E.g. licensing needs to be configured, any desired plugins will need to be installed, etc.
2. **Set the amount of memory** The installer investigates the amount of RAM on the machine during installation and sets the amount of memory that the Workbench can use.

3. **Establish shortcuts (optional)** On Windows and Mac systems, an option is provided during installation to create a shortcut for starting the Workbench. On Linux systems, this option is also presented, but it has no effect.

Updating workflows after upgrading a *CLC Workbench* is described in section 11.5.1.

### 1.2.2 Installation on Microsoft Windows

To install to a central location, the installer must be run in *administrator mode*. On Windows, right-click on the of the downloaded installer file and choose "Run as Administrator".

To install to a location you have write permission to, double click the icon of the downloaded installer file.

Then walk through the following steps. (The exact order options are presented may differ to that described.)

- Read and accept the License agreement and click **Next**.
- Unless you are installing a minor update to the same folder as an existing installation, you will be prompted to choose where you would like to install the Workbench. If you are upgrading from an earlier version, please refer to section 1.2.1 for information about installing to an existing or different directory. Click on **Next**.
- Choose where you want the program’s shortcuts to be placed. Click on **Next**.
- Choose if you would like to associate .clc files to the *CLC Main Workbench*. If you check this option, double-clicking a file with a "clc" extension will open the *CLC Main Workbench*.
- Choose if a desktop icon should be created, and choose whether clc://URLs should be opened by this program by default. Click on **Next**.
- Wait for the installation process to complete, and then choose whether you would like to launch *CLC Main Workbench* right away. Click on **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

To install to a location you have write permission to, double click the icon of the downloaded installer file.

Then walk through the following steps. (The exact order options are presented may differ to that described.)

- Read and accept the License agreement and click **Next**.
Choose where you would like to install the application. If you are upgrading from an earlier version, please refer to section 1.2.1 for information about installing to an existing or different directory. Click on Next.

Specify other options associated with the installation such as whether a desktop icon should be created, whether the software should open clc:// URLs, whether .clc files should be associated with the software and whether it should be added to the dock. Click on Next.

Wait for the installation process to complete, choose whether you would like to launch CLC Main Workbench right away, and click on Finish.

When the installation is complete, the program can be launched from the dock, if present there, or by clicking on the desktop shortcut if you chose to create one. The software can also be launched from within the installation folder.

1.2.4 Installation on Linux with an installer

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

```
# sh CLCMainWorkbench_23_0_5_64.sh
```

To install to a central location such as /opt or /usr/local, you will normally need to run the above command using sudo. If you do not have sudo privileges you can choose to install in your home directory, or any other location you have write permission for.

Then walk through the following steps. (The exact order options are presented may differ to that described.)

- Read and accept the License agreement and click Next.
- Choose where you would like to install the application. If you are upgrading from an earlier version, please refer to section 1.2.1 for information about installing to an existing or different directory. Click on Next.
- Choose where you would like to create symbolic links to the program. Click on Next. **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 on 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:

```
# clcmainwb23
```
Otherwise you start the application by navigating to the location where you choose to install it and running the command:

```
# ./clcmainwb23
```

### 1.3 System requirements

The system requirements of CLC Main Workbench are:

- Mac: macOS 11, 12, and 13
- Linux: RHEL 7 and later, SUSE Linux Enterprise Server 12 and later. The software is expected to run without problem on other recent Linux systems, but we do not guarantee this. To use BLAST related functionality, libnsl.so.1 is required.
- 64 bit operating system
- 1 GB RAM required
- 2 GB RAM recommended
- 1024 x 768 display required
- 1600 x 1200 display recommended

**System requirements for 3D viewers**

A graphics card that supports OpenGL 2.0.

**Note:** 3D rendering is only supported when the CLC Main Workbench is installed on the same machine the viewer is opened on. Indirect rendering (such as X11 forwarding through ssh), remote desktop connection/VNC, and running in virtual machines is not supported.

**CLC data location requirements**

Requirements for CLC data locations are provided in section 3.1.2.

### 1.3.1 Limitations on maximum number of cores

Most modern CPUs implement hyper threading or a similar technology that makes each physical CPU core appear as two logical cores on a system. In this manual the term "core" always refers to a logical core unless otherwise stated.

**Licensing:** Static licenses can be used on systems with 64 or fewer logical cores. On systems with more than 64 logical cores a network license is needed. See [https://digitalinsights.qiagen.com/licensing/](https://digitalinsights.qiagen.com/licensing/).
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 1.2, 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:

![License Manager](image)

You need a license...
In order to use the application you need a valid license. Please choose how you would like to obtain a license for your Workbench.

- **Request an evaluation license**
  Try out the application for 14 days. A static license will be downloaded to your local machine. Use with remote or virtual machines is not supported.

- **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 a static license from an existing license file.

- **Upgrade from an existing Workbench installation**
  Upgrade an existing license for an older version of the software. Your license must be covered by Maintenance, Upgrades and Support to use this option.

- **Configure license manager connection**
  Configure a connection to a CLC Network License Manager or CLC License Server that hosts network license(s) for this product, or update or disable an existing connection configuration.

![Figure 1.2: The License Assistant provides access to licensing options.](image)

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 Network License Manager or 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 free of charge for a 14 day period for evaluation purposes. The 14 day period commences when the evaluation license is downloaded. If you have questions about CLC Main Workbench features or product licensing options, please send an email to bioinformaticssales@qiagen.com.

When you choose the option Request an evaluation license, you will see the dialog shown in figure 1.3.

![Request an evaluation license dialog](image)

Figure 1.3: 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 1.4 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.
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 1.5.

![License download form](image)

**Figure 1.5: 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 1.6.

![License import dialog](image)

**Figure 1.6: 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. 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).

![Figure 1.7: Enter a license order ID into the text field and then click on the Next button.](image)

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.

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.
CHAPTER 1. INTRODUCTION TO CLC MAIN WORKBENCH

Requesting a license...

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

Download a license

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.

Accepting the license agreement

The End User License Agreement (EULA) must be read and accepted as part of 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 1.11.

![Figure 1.11: Selecting a license file.](image)

Click on the **Choose License File** button, locate the license file and selected 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. 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 1.12.

When you click on the **Next** button, the Workbench checks if you are entitled to upgrade your
CHAPTER 1. INTRODUCTION TO CLC MAIN WORKBENCH

Figure 1.12: An license from an older installation was found.

Figure 1.13: An license from an older installation was found.

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 1.13 will be displayed. If there is a problem updating the selected license, a dialog will appear indicating this.

In this dialog, there are two options:

- **Direct Download.** Download the license directly. This method requires that the Workbench

---

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.
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 1.14 will appear if the license is successfully downloaded and installed.

![Requesting a license](image)

Figure 1.14: 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 1.15.

![Download a license](image)

Figure 1.15: 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 1.16.

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. 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 manager connection**

If your organization is running a **CLC Network License Manager** or 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 Network License Manager** or CLC License Server, select the **Configure License Server connection** option and click on the **Next** button. A dialog appears, as shown in figure 1.17.

Figure 1.17: **Connecting to a CLC Network License Manager or CLC License Server.**
The options in that dialog are:

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

- **Automatically detect license manager.** By checking this option the Workbench will look for a *CLC Network License Manager* or 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 Network License Manager* or CLC License Server using the **Manually specify license manager** option instead.

- **Manually specify license manager.** Select this option to enter the details of the machine the *CLC Network License Manager* or CLC License Server software is running on, specifically:
  - **Host name.** The address of the machine the *CLC Network License Manager* or CLC License Server software is running on.
  - **Port.** The port used by the *CLC Network License Manager* or 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 used when requesting a network license 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**

A valid module license is needed to start a module tool, or a workflow including a module tool. Network licenses for modules are 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 Network License Manager* or 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 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 Network License Manager* or *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 1.18.

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.

5. Click **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 request another license.

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 1.19 when you start up the Workbench.

![Figure 1.19: This window appears when there are no available network licenses for the software you are running.](image)

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 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.

• Lost connection to the CLC Network License Manager or CLC License Server If the Workbench connection to the CLC Network License Manager or CLC License Server is lost, you will see a dialog like that shown in figure 1.20.

![Figure 1.20: Here, the Workbench is unable to establish a connection to obtain a network license.](image)

If you have chosen the option to Automatically detect license manager and you have not succeeded in connecting to the CLC Network License Manager or 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 settings, as described earlier in this section.

If you have successfully contacted the CLC Network License Manager or CLC License Server from your Workbench previously, please consider discussing this issue with your administrator, for example, making sure that the CLC Network License Manager or 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:
The license manager is shown in figure 1.21.

![License Manager](image)

Figure 1.21: The License Manager provides information about licenses being used and access to other license-related functionality.

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 Network License Manager or CLC License Server. Click on the Configure Network License button at the lower left corner to open the dialog seen in figure 1.17.
- Upgrade from an evaluation license. Click on the Upgrade Workbench License button to open the dialog shown in figure 1.2.
- 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:

- 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 page:\footnote{For CLC Genomics Workbench 5.x and earlier or CLC Main Workbench 6.7.x and earlier, the license download page URL is \url{http://licensing.clcbio.com/LmxWSv1/GetLicenseFile}}

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.

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 1.22.

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 1.23. This button is also visible in message windows that appear if a Workbench is started up that has an expired license.
Figure 1.22: **Bioinformatics tools available when using Viewing Mode are found in the Toolbox.**

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.

Figure 1.23: **Click on the Viewing Mode button at the bottom of the License Manager window to launch 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.

### 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

The functionality of the **CLC Main Workbench** can be extended by installing plugins. The built-in **Plugin Manager** provides an up-to-date listing of the plugins available.

Alternatively, visit our plugin webpage for a list: [https://digitalinsights.qiagen.com/products-overview/plugins/](https://digitalinsights.qiagen.com/products-overview/plugins/).

Plugins are installed and uninstalled using the **Plugin Manager**, which can be opened using the **Plugins** (🔧) button in the **Toolbar**, or by going to the top level menu:

**Utilities | Manage Plugins... (🔧)**

**Note**: To install plugins and modules using a centrally installed **CLC Workbench**, the software must be run in **administrator mode**. On Windows, right-click on the program shortcut and choose "Run as Administrator". On Linux, this usually means running the software with sudo privileges.

The **Plugin Manager** has two tabs at the top:

- **Manage Plugins**: An overview of plugins that are installed.
- **Download Plugins**: An overview of plugins available from QIAGEN that are not yet installed on your Workbench.

### 1.5.1 Install

To install a plugin, open the **Plugin Manager** and click on the **Download Plugins** tab. This will display an overview of the plugins available (figure 1.24).

Select a plugin in the list to display additional information about it in the right hand pane. Click on **Download and Install** to install the plugin.

**Accepting the license agreement**

The **End User License Agreement (EULA)** must be read and accepted as part of 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 you have a .cpa plugin installer file on your computer, for example if you have downloaded it from our website, install the plugin by clicking on the **Install from File** button at the bottom of the
dialog and specifying the plugin *.cpa file.

When you close the Plugin Manager after making changes, you will be prompted to restart the software. Plugins will not be fully installed, or removed, until the CLC Workbench has been restarted.

### 1.5.2 Uninstall

Plugins are uninstalled using the Plugin Manager (figure 1.25). This can be opened using the **Plugins ( )** button in the Toolbar, or by going to the top level menu:

**Utilities | Manage Plugins... ( )**

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 (figure 1.26).

![Plugin updates](image)

**Figure 1.26: Plugin updates.**

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 1.25).

### 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, go to **Edit | Preferences** and choose the **Advanced** tab (figure 4.9).

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

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

List hosts that should be contacted directly, i.e. not via the proxy server, in the **Exclude hosts** field. The value can be a list, with each host separated by a | symbol. The wildcard character * can also be used. For example: *.foo.com|localhost.

The proxy can be bypassed when connecting to a **CLC Server**, as described in section 6.1.

If you have any problems with these settings, contact your systems administrator.
CHAPTER 1. INTRODUCTION TO CLC MAIN WORKBENCH

1.7 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.

Figure 1.27: Adjusting proxy settings.
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 Close views ....................................... 44
  2.1.2 Save changes in a view ........................... 44
  2.1.3 Undo/Redo .......................................... 45
  2.1.4 Arrange views in View Area .................... 46
  2.1.5 Moving a view to a different screen .......... 47
  2.1.6 Side Panel ......................................... 47
2.2 Zoom functionality in the View Area ............. 49
2.3 Toolbox and Favorites tabs .......................... 51
  2.3.1 Toolbox tab ....................................... 51
  2.3.2 Favorites tab ..................................... 51
2.4 Processes tab and Status bar ....................... 54
2.5 History and Element Info views ................... 55
2.6 Workspace .............................................. 56
2.7 List of shortcuts ....................................... 58

The user interface of the CLC Main Workbench when it is first opened looks like that shown in figure 2.1.

Key areas are listed below with a brief description and links to further information.

- **Navigation Area** Data elements stored in File Locations are listed in the Navigation Area. (Section 3.1).

- **Toolbox Area** This area contains 3 tabs:
  - **Processes** Running and finished processes are listed under this tab. (Section 2.4)
  - **Toolbox** Analysis tools and installed workflows are listed and can be launched from under this tab. (Section 2.3.1)
  - **Favorites** Tools you use most often are listed here, and you can add tools you want, for quick access. (Section 2.3.2)
• **View Area** Data and workflow designs can be opened in this area for viewing and editing. (Section 2.1) When elements are open in the View Area, a Side Panel with configuration options will be present on the right hand side. (Section 4.6)

• **Menu bar** and **Tool bar** Many tools and associated actions can be launched using buttons and options in these areas.

• **Status Bar** The Workbench status and its connections to other systems is presented in this area. (Section 2.4)

![Figure 2.1: The CLC Workbench interface includes the Navigation Area in the top left, several tabs in the Toolbox area at the bottom left, a large viewing area on the right, menus and toolbars at the top, and a status bar at the bottom.](image)

Different areas of the interface can be hidden or made visible, as desired. Options controlling this are available under the View menu at the top. For example, what is shown in the Toolbox area can be configured using the menu options found under:

**View | Show/Hide Toolbox**

You can also collapse the various areas by clicking on buttons like (��) or (显示), where they appear. Similar buttons are presented for revealing areas if they are hidden.

### 2.1 View Area

The **View Area** is where elements are displayed when opened. Each open element is shown in a view, with a tab at the top containing the element’s name (figure 2.2). Mouse-over a tab to reveal a tooltip containing summary information about the element.

Tabs can be dragged to put them in the order desired, and to move them to a different area in a split view (section 2.1.4).
Right-clicking on a tab opens a menu with various navigation options, as well as the ability to select tools and viewing options, etc.

**Opening and viewing elements in the View Area**

There are multiple ways to open an element in the View Area, including:

1. Double click on an element in the Navigation Area.
2. Right-click on an element in the Navigation Area, and choose the **Show** option from the context menu.
3. Drag elements from the Navigation Area into the Viewing Area.
4. Select an element in the Navigation Area and use the keyboard shortcut Ctrl + O (⌘ + O on macs)
5. Choose the option to "Open" results when launching an analysis. (This is only recommended when small numbers of elements will be generated, and where it is not important to save the results directly.)

When opening an element while another element is already open, the newly opened element will become the active tab. Click on any other tab to open it and make it the active view. Alternatively, use the keyboard shortcuts to navigate between tabs: Ctrl + PageUp or PageDown (or ⌘ + PageUp or PageDown on macs).

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.

**Tooltips**

For some data types and some views, tooltips provide additional useful information. Hover the mouse cursor over an area of interest to reveal these. For example, hover over an annotation on a sequence and a tooltip containing details about that annotation is shown. Hover over a variant in a variant track, and information about that variant is shown.

If you wish to hide such tooltips while moving the mouse around in a view, hold down the Ctrl key. Tooltips can take a moment to appear. To make them show up immediately while moving the mouse around in a view, hold down the Shift key.

**Accessing different views of the same data element**

Various views of a data element are available. These can be opened by clicking on the icons at the bottom of an open view. The last two icons are always those for opening the **History** (🗹) and **Element Info** (🔍) views. These are described in section 2.5.

For illustration, the icons for views available for sequence elements are shown in figure 2.3. Clicking on the **Show As Circular** ((circle)) icon would present the sequence in a circular view. Mouse over any of these icons to see the type of view they represent.
Figure 2.2: Four elements are open in the View Area, organized in 2 areas horizontally - 3 elements in the top area, and one in the bottom. The active view is in the top area, as indicated by the blue bar just under the tabs.

Figure 2.3: The icons presented at the bottom of an open nucleotide sequence. Clicking on each of these presents a different view of the data.

Linked views

More than one view of a given element can be open at the same time by taking advantage of multiple viewing areas, i.e. split views. To open an additional view of the same data element, keep the Ctrl key (⌘ on macs) depressed and then click on the icon for the additional view you wish to open. This will open the new view in a second, horizontal view area. If the View Area is already split horizontally, the new view will be opened in the area not occupied by the original view.

The two open views are linked. For example, selecting a region of sequence in one view would also select it in the other view. An example is shown in figure 2.4, where a region was selected in one view, and that selection is then also shown in the other view.

The linked nature of multiple views can also pertain to separate data elements in some cases. These are described in relevant areas of the manual.

Further information about split views is provided in section 2.1.4.
2.1.1 Close views

To close a view, right-click on the element tab and choose the Close option suited to your needs. See figure 2.5.

- **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.

2.1.2 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: The ATP8a1 mRNA element has been edited, but the changes are not saved yet. This is indicated by an * on the tab name in the View Area, and by the use of bold, italic font for the element’s name 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 (Ctrl+) 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.

2.1.3 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 (\(^{(\text{undo})}\)) 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 3.1.8).

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

2.1.4 Arrange views in View Area

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

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).

A blue bar underneath the tab area indicates the active area, i.e. where the focus is at that time.

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**)  
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**

### 2.1.5 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.](image)

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.

### 2.1.6 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**.

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 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 (→).

To show or hide the Side Panel:

- Click the (↑) at the top right corner of the Side Panel to hide | Click the (↓) to the right to show

or use the keyboard shortcut Ctrl + U

At the bottom of the Side Panel (see figure 2.12) there are a number of icons used to:

- Collapse all settings (←).
- Expand all settings (→).
- Dock all palettes (□).
- Get Help for the particular view and 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.

- 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 functionality in the View Area

All views except tabular and text views support zooming in and out. This section describes the most common zoom functionality. For protein 3D structures, see section 13.2.

Zoom tools are located at the bottom right corner of most views (figure 2.13).

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

- Shortcuts for zooming out to fit the width of the view (\[H\]) or zooming in all the way to see details (\[T\]).

- A shortcut to zoom to a selection (\[B\]). Select a region in the view, and then click this icon to zoom in on the selected region. (Keyboard shortcut Ctrl + 1)

- A slider to zoom in and zoom out to any desired level. The slider position reflects the current zoom level. Move the slider left to zoom out, or right to zoom in. For fine grained control, click on the slider and move the mouse up slightly or down slightly.

- Mouse mode buttons:
  - **Selection mode** (\[C\]). Used when you wish to select data in a view. This is the default.
  - **Zoom in mode** (\[V\]) When selected, whenever you click the view, it zooms in. Alternatively, click on a location in the view, and the view will zoom in, with the focus on that location, or drag a box around an area, and the view will be zoomed to that area. (Keyboard shortcut Ctrl + 2)
    If you press and hold on (\[R\]) or right-click on it, two other modes become available (figure 2.14).
  - **Panning** (\[P\]) When selected, you can pan around in the view using the mouse. (Keyboard shortcut Ctrl + 4)
  - **Zoom out** (\[S\]) When selected, whenever you click the view, it zooms out. (Keyboard shortcut Ctrl + 3)

![Figure 2.14](image)

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

Additional notes:

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

- If you press the **Shift** button on your keyboard while in zoom mode, the zoom function is reversed.

- You may have to click in the view before you can use the keyboard or the scroll wheel to zoom.

In many views, you can zoom in by pressing '+' on your keyboard, or zoom out by pressing '-' on your keyboard.

If you have a mouse with a scroll wheel, you can also do the following:
2.3 Toolbox and Favorites tabs

2.3.1 Toolbox tab

The Toolbox tab contains tools and installed workflows, including those distributed via plugins, as well as external applications configured and enabled on a CLC Genomics Server that the Workbench is connected to. See figures 2.15 and 2.16.

Launching analyses from the Toolbox can be done by:

- Double-clicking on the tool or workflow name.
- Right-clicking on the tool or workflow name and choosing the option "Run" from the menu that appears.
- Dragging elements from the Navigation Area onto the name of a tool or workflow.

Other methods of launching tools, including using the Quick Launch tool ( ), are described in section 9.1.

Figure 2.15: The Toolbox tab in the bottom right contains folders of available tools, and when available, installed workflows. This Workbench is not connected to a CLC Server, as indicated by the grey server icon in the status bar.

2.3.2 Favorites tab

You can specify tools, or folders of tools, that you want to find quickly as favorites. In addition, the 10 tools you use most frequently are automatically identified as your frequently used tools. Favorites and frequently used tools are made available under the Favorites tab, which is available in three locations:

- In the bottom, left side of the Workbench in the Toolbox area (figure 2.17)
Figure 2.16: This Workbench is connected to a CLC Server, as indicated by the blue server icon in the status bar. External applications have been configured and enabled on that CLC Server, so an External Applications folder is listed, which contains those external applications. The server icon within that folder’s icon is a reminder that these are only available when logged into the CLC Server.

- In the Launch wizard, which is started using the Launch button in the top Toolbar.
- In the Add Elements dialog available when you are creating or editing a workflow.

Figure 2.17: Under the Favorites tab is a folder containing your frequently used tools and a folder containing tools you have specified as your favorites.

To manually add tools to the Favorites tab, go to the the Toolbox area in the bottom, left hand side of the Workbench and:

- Right-click on a tool or folder of tools in the Toolbox tab and choose the option "Add to Favorites" from the menu that appears (figure 2.18), or
- Open the Favorites tab, right-click in the Favorites folder, choose the option "Add tools" or "Add group of tools". Then select the tool or tool group to add.
- From the Favorites tab, click on a tool in the Frequently used folder and drag it into the Favorites folder.

Tools within the Favorites folder can be re-ordered by opening the Favorites tab in the Toolbox area in the bottom, left hand side of the Workbench and dragging tools up and down within the list. (Folders cannot be repositioned.)
Figure 2.18: Tools manually added to the Favorites tab are listed at the top. Tools under the "Frequently used" section are added automatically, based on usage.

To remove a tool from the Favorites tab, right-click on it and choose the option Remove from Favorites from the menu that appears.
CHAPTER 2. USER INTERFACE

2.4 Processes tab and Status bar

The Status bar is located at the bottom of the CLC Main Workbench. On the left side, information is displayed about whether a process is running or the Workbench is idle. Further to the left is information on the status of connections to other systems, such as a CLC Server. On the right hand side, context dependent information is displayed. For example, when selecting part of a sequence, the size of a selected region will be reported, or when mousing over a variant in a variant track, the location of the variant is reported.

Detailed information about running and completed processes for the Workbench session is provided under the Processes tab, found in the lower, left side of the Workbench. When logged into a CLC Server, the status of your jobs that are running, completed or queued on the server, are also displayed.

Several options are available after clicking on the small icon ( ) next to a given process, as shown in figure 2.19).

Figure 2.19: A database search and an alignment calculation are running. Clicking the small icon next to the process lists actions you can take for that process.

For completed jobs, these options provide a convenient way to locate results in the Navigation Area:

- **Show results** Open the results generated by that process in the Viewing Area. (Relevant if results were saved, as described in section 9.2.)
- **Find results** Highlight the results in the Navigation Area. (Relevant if results were saved, as described in section 9.2.)
- **Show Log Information** 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** Show any messages that were produced during the processing of your data.

Stopped, paused and finished processes are not automatically removed from the Processes tab during a Workbench session. They can, however, 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:

Utilities | Remove Finished Processes (X)

If you close the Workbench while jobs are still running on it, a dialog will ask for confirmation before closing. Workbench processes are stopped when the software is closed and these
processes are not automatically restarted when you start the Workbench again. Closing the Workbench does *not* interrupt jobs sent to a CLC Server, as described below.

**Processes submitted to a CLC Server**

Processes 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.

### 2.5 History and Element Info views

**History view**

To open the History view, click on the Show History (§) icon under the View area.

The History view shows the log of all operations carried out on this element. This detailed record can be viewed within the CLC Main Workbench, as described here, or exported to a pdf format file.

The table at the top of the History view contains a row for each operation that has affected this data element. When rows are selected in the table, full details for those operations are displayed in the bottom panel (figure 2.20).

The summary information shown in the table for each operation is:

- **Description** The operation performed
- **User** The username of the person who performed the operation. If you import data created by another person in a CLC Workbench, that person’s username will be shown.
- **Date and time** Date and time the operation was carried out. These are displayed according to your locale settings (see section 4.1).
- **Version** The software name and version used for that operation.
In addition to the fields above, the following fields are displayed in the lower panel:

- **Parameters** The parameter values used for an analysis.
- **Comments** Additional details added here by tools or details that have been added manually. Click on *Edit* to add information to this field.
- **Originates from** The elements that the current element originates from. Clicking the name of an element here selects it in the Navigation Area. Clicking the "history" link opens that element with its History view shown.

If the element was created as a result of running a workflow, the name and version of the workflow is sown in the side panel under the Workflow details tab.

**Element Info view**

To open the Element Info view, click on the *Show Element Info* icon under the View area.

The Element Info view contains information about the element, such as its name, description and other attributes. If the element is associated with metadata, that association is also reported here.

For further details about element information, please see section 12.4. For further information about metadata associations, see section 10.3.2.

### 2.6 Workspace

Workspaces are useful when working on more than one project. Open views and their arrangement are saved in a given workspace. Switching between workspaces can thus save much time when working on several different sets of data and results.

Initially, there is a single workspace called “Default”. When you set up other workspaces, you assign each a name, which is used when re-opening that workspace, and which is displayed in the title bar of the Workbench when it is the active workspace.
The state of each workspace is saved automatically when the Workbench is closed down. The workspace that was open when closing down is the one that will be opened when the Workbench is started up again.

Figure 2.21: The workspace called "My Microbial Workspace" is open after selecting it from the Workspaces menu using the button in the Toolbar. The name of the workspace is visible in the Workbench title bar.

Workspaces do not affect the underlying organization of data, so folders and elements remain the same in the Navigation Area.

Workspaces can be created, opened and deleted using the options available under the Workspace button in the top Toolbar. The same menu is also available from under the View menu, by selecting the Workspaces option there. In the instructions below, we focus on using the Toolbar button.

Creating a workspace

Create a new workspace by clicking in the Workspace button in the top Toolbar.

In the drop-down menu that appears, choose the option Create Workspace.

In the dialog that appears, enter a name for the new workspace.

Upon creation, you are in the new Workspace. The name of the workspace will be in the title bar of the Workbench.

Initially, the Navigation Area may be collapsed. Open it up again by clicking in the small black triangle at the top right of the Navigation Area.

View Area is empty and ready to work with.

Opening a workspace

Switch between workspaces by clicking in the Workspace button in the top Toolbar and selecting the desired workspace from the list presented.
The name of the active workspace will be greyed out in the list.

**Deleting a workspace**

To delete a workspace, click on the **Workspace** button in the top Toolbar and select the option **Delete Workspace**.

Workspaces that can be deleted are listed in a drop-down menu in the dialog that appears. Select the one to delete.

Deletion of workspaces cannot be undone.

Note: The Default workspace is not offered, as it cannot be deleted.

### 2.7 List of shortcuts

The keyboard shortcuts available in *CLC Main Workbench* are listed below.
<table>
<thead>
<tr>
<th>Action</th>
<th>Windows/Linux</th>
<th>macOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adjust selection</td>
<td>Shift + arrow keys</td>
<td>Shift + arrow keys</td>
</tr>
<tr>
<td>Back to Navigation Area</td>
<td>Alt + Home or Alt + fn + left arrow</td>
<td>⌘ + Home or ⌘ + fn + left arrow</td>
</tr>
<tr>
<td>BLAST</td>
<td>Ctrl + Shift + L</td>
<td>⌘ + Shift + L</td>
</tr>
<tr>
<td>BLAST at NCBI</td>
<td>Ctrl + Shift + B</td>
<td>⌘ + Shift + B</td>
</tr>
<tr>
<td>Close</td>
<td>Ctrl + W</td>
<td>⌘ + W</td>
</tr>
<tr>
<td>Close all views</td>
<td>Ctrl + Shift + W</td>
<td>⌘ + Shift + W</td>
</tr>
<tr>
<td>Copy</td>
<td>Ctrl + C</td>
<td>⌘ + C</td>
</tr>
<tr>
<td>Create alignment</td>
<td>Ctrl + Shift + A</td>
<td>⌘ + Shift + A</td>
</tr>
<tr>
<td>Create track list</td>
<td>Ctrl + L</td>
<td>⌘ + L</td>
</tr>
<tr>
<td>Cut</td>
<td>Ctrl + X</td>
<td>⌘ + X</td>
</tr>
<tr>
<td>Delete</td>
<td>Delete</td>
<td>Delete or ⌘ + Backspace</td>
</tr>
<tr>
<td>Exit</td>
<td>Alt + F4</td>
<td>⌘ + Q</td>
</tr>
<tr>
<td>Export</td>
<td>Ctrl + E</td>
<td>⌘ + E</td>
</tr>
<tr>
<td>Export graphics</td>
<td>Ctrl + G</td>
<td>⌘ + G</td>
</tr>
<tr>
<td>Find Next Conflict</td>
<td>'.' (dot)</td>
<td>'.' (dot)</td>
</tr>
<tr>
<td>Find Previous Conflict</td>
<td>',' (comma)</td>
<td>',' (comma)</td>
</tr>
<tr>
<td>Help</td>
<td>F1</td>
<td>F1</td>
</tr>
<tr>
<td>Import</td>
<td>Ctrl + I</td>
<td>⌘ + I</td>
</tr>
<tr>
<td>Launch tools</td>
<td>Ctrl + Shift + T</td>
<td>⌘ + Shift + T</td>
</tr>
<tr>
<td>Maximize/restore View size</td>
<td>Ctrl + M</td>
<td>⌘ + M</td>
</tr>
<tr>
<td>Move gaps in alignment</td>
<td>Ctrl + arrow keys</td>
<td>⌘ + arrow keys</td>
</tr>
<tr>
<td>New Folder</td>
<td>Ctrl + Shift + N</td>
<td>⌘ + Shift + N</td>
</tr>
<tr>
<td>New Sequence</td>
<td>Ctrl + N</td>
<td>⌘ + N</td>
</tr>
<tr>
<td>Panning Mode</td>
<td>Ctrl + 4</td>
<td>⌘ + 4</td>
</tr>
<tr>
<td>Paste</td>
<td>Ctrl + V</td>
<td>⌘ + V</td>
</tr>
<tr>
<td>Print</td>
<td>Ctrl + P</td>
<td>⌘ + P</td>
</tr>
<tr>
<td>Redo</td>
<td>Ctrl + Y</td>
<td>⌘ + Y</td>
</tr>
<tr>
<td>Rename</td>
<td>F2</td>
<td>F2</td>
</tr>
<tr>
<td>Save</td>
<td>Ctrl + S</td>
<td>⌘ + S</td>
</tr>
<tr>
<td>Save As</td>
<td>Ctrl + Shift + S</td>
<td>⌘ + Shift + S</td>
</tr>
<tr>
<td>Scrolling horizontally</td>
<td>Shift + Scroll wheel</td>
<td>Shift + Scroll wheel</td>
</tr>
<tr>
<td>Search local data</td>
<td>Ctrl + Shift + F</td>
<td>⌘ + Shift + F</td>
</tr>
<tr>
<td>Search via Side Panel</td>
<td>Ctrl + F</td>
<td>⌘ + F</td>
</tr>
<tr>
<td>Search NCBI</td>
<td>Ctrl + B</td>
<td>⌘ + B</td>
</tr>
<tr>
<td>Search UniProt</td>
<td>Ctrl + Shift + U</td>
<td>⌘ + Shift + U</td>
</tr>
<tr>
<td>Select All</td>
<td>Ctrl + A</td>
<td>⌘ + A</td>
</tr>
<tr>
<td>Select Selection Mode</td>
<td>Ctrl + 1 (one)</td>
<td>⌘ + 1 (one)</td>
</tr>
<tr>
<td>Show folder content</td>
<td>Ctrl + O</td>
<td>⌘ + O</td>
</tr>
<tr>
<td>Show/hide Side Panel</td>
<td>Ctrl + U</td>
<td>⌘ + U</td>
</tr>
<tr>
<td>Sort folder</td>
<td>Ctrl + Shift + R</td>
<td>⌘ + Shift + R</td>
</tr>
<tr>
<td>Split Horizontally</td>
<td>Ctrl + T</td>
<td>⌘ + T</td>
</tr>
<tr>
<td>Split Vertically</td>
<td>Ctrl + J</td>
<td>⌘ + J</td>
</tr>
<tr>
<td>Switch tabs in View Area</td>
<td>Ctrl + PageUp/PageDown</td>
<td>Ctrl + PageUp/PageDown</td>
</tr>
<tr>
<td>or Ctrl + fn + arrow up/down</td>
<td>or Ctrl + fn + arrow up/down</td>
<td></td>
</tr>
<tr>
<td>Switch views</td>
<td>Ctrl + Shift + PageUp</td>
<td>Ctrl + Shift + PageUp</td>
</tr>
<tr>
<td>Ctrl + Shift + PageDown</td>
<td>Ctrl + Shift + PageDown</td>
<td>Ctrl + Shift + PageDown</td>
</tr>
<tr>
<td>Translate to Protein</td>
<td>Ctrl + Shift + P</td>
<td>⌘ + Shift + P</td>
</tr>
<tr>
<td>Undo</td>
<td>Ctrl + Z</td>
<td>⌘ + Z</td>
</tr>
<tr>
<td>Update folder</td>
<td>F5</td>
<td>F5</td>
</tr>
<tr>
<td>User Preferences</td>
<td>Ctrl + K</td>
<td>⌘ + ,</td>
</tr>
</tbody>
</table>
### Scroll and Zoom shortcuts

<table>
<thead>
<tr>
<th>Action</th>
<th>Windows/Linux</th>
<th>macOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertical scroll in reads tracks</td>
<td>Alt + Scroll wheel</td>
<td>Alt + Scroll wheel</td>
</tr>
<tr>
<td>Vertical scroll in reads tracks, fast</td>
<td>Shift+Alt+Scroll wheel</td>
<td>Shift+Alt+Scroll wheel</td>
</tr>
<tr>
<td>Vertical zoom in graph tracks</td>
<td>Ctrl + Scroll wheel</td>
<td>?? + Scroll wheel</td>
</tr>
<tr>
<td>Zoom</td>
<td>Ctrl + Scroll wheel</td>
<td>?? + Scroll wheel</td>
</tr>
<tr>
<td>Zoom In Mode</td>
<td>Ctrl + 2</td>
<td>?? + 2</td>
</tr>
<tr>
<td>Zoom In (without clicking)</td>
<td>'+' (plus)</td>
<td>'+' (plus)</td>
</tr>
<tr>
<td>Zoom Out Mode</td>
<td>Ctrl + 3</td>
<td>?? + 3</td>
</tr>
<tr>
<td>Zoom Out (without clicking)</td>
<td>'-' (minus)</td>
<td>'-' (minus)</td>
</tr>
<tr>
<td>Zoom to base level</td>
<td>Ctrl + 0</td>
<td>?? + 0</td>
</tr>
<tr>
<td>Zoom to fit screen</td>
<td>Ctrl + 6</td>
<td>?? + 6</td>
</tr>
<tr>
<td>Zoom to selection</td>
<td>Ctrl + 5</td>
<td>?? + 5</td>
</tr>
<tr>
<td>Reverse zoom mode</td>
<td>press and hold Shift</td>
<td>press and hold Shift</td>
</tr>
</tbody>
</table>

### Workflows related shortcuts

<table>
<thead>
<tr>
<th>Action</th>
<th>Windows/Linux</th>
<th>macOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adjust workflow layout</td>
<td>Shift + Alt + L</td>
<td>?? + Shift + Alt + L</td>
</tr>
<tr>
<td>Workflow, add element</td>
<td>Alt + Shift + E</td>
<td>Alt + Shift + E</td>
</tr>
<tr>
<td>Workflow, collapse if its expanded</td>
<td>Alt + Shift + '-' (minus)</td>
<td>Alt + Shift + '-'</td>
</tr>
<tr>
<td>Workflow, create installer</td>
<td>Alt + Shift + I</td>
<td>Alt + Shift + I</td>
</tr>
<tr>
<td>Workflow, execute</td>
<td>Ctrl + enter</td>
<td>?? + enter</td>
</tr>
<tr>
<td>Workflow, expand if its collapsed</td>
<td>Alt + Shift + '+' (plus)</td>
<td>Alt + Shift + '+'</td>
</tr>
<tr>
<td>Workflow, highlight used elements</td>
<td>Alt + Shift + U</td>
<td>Alt + Shift + U</td>
</tr>
<tr>
<td>Workflow, remove all elements</td>
<td>Alt + Shift + R</td>
<td>Alt + Shift + R</td>
</tr>
</tbody>
</table>

### Combinations of keys and mouse movements

<table>
<thead>
<tr>
<th>Action</th>
<th>Windows/Linux</th>
<th>macOS</th>
<th>Mouse movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximize View</td>
<td></td>
<td></td>
<td>Double-click the tab of the View</td>
</tr>
<tr>
<td>Restore View</td>
<td></td>
<td></td>
<td>Double-click the View title</td>
</tr>
<tr>
<td>Reverse zoom mode</td>
<td>Shift</td>
<td>Shift</td>
<td>Click in view</td>
</tr>
<tr>
<td>Select multiple elements not grouped together</td>
<td>Ctrl</td>
<td>??</td>
<td>Click elements</td>
</tr>
<tr>
<td>Select multiple elements grouped together</td>
<td>Shift</td>
<td>Shift</td>
<td>Click elements</td>
</tr>
<tr>
<td>Select Editor and highlight the corresponding element in the Navigation Area</td>
<td>Alt or Ctrl</td>
<td>??</td>
<td>Click tab</td>
</tr>
</tbody>
</table>

*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 .......................................................... 61
  3.1.1 Data structure ................................................. 63
  3.1.2 Adding locations ............................................. 64
  3.1.3 Create new folders ......................................... 66
  3.1.4 Sorting folders .................................................. 67
  3.1.5 Multiselecting elements .................................. 67
  3.1.6 Moving and copying elements ......................... 67
  3.1.7 Change element names ................................... 68
  3.1.8 Delete, restore and remove elements ................. 69
  3.1.9 Show folder elements in a table ..................... 70
3.2 Working with tables .................................................... 71
  3.2.1 Filtering tables ............................................. 73
3.3 Customized attributes on data locations ....................... 75
  3.3.1 Filling in values .......................................... 77
  3.3.2 What happens when a clc object is copied to another data location? ... 79
  3.3.3 Searching custom attributes ......................... 79
3.4 Searching for data in CLC Locations ................................. 79
  3.4.1 Quick Search ............................................. 80
  3.4.2 Local Search ............................................. 82

This chapter explains general 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 for data in your CLC Locations. The use of metadata tables in CLC Main Workbench is described separately, in chapter 10.

3.1 Navigation Area

The Navigation Area (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.
Each data element has a name and an icon. The icon represents the type of data in the element. A list of many of the icons and the type of data they represent can be found in our FAQ at https://qiagen.my.salesforce-sites.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA4li000000L5uFCAS.

Just above the area with the listing of data are 4 icons that can be clicked upon. From left to right, these are:

- **Collapse all** ( ). Close all the open folders in the Navigation Area.
- **Add File Location** ( ). Add a new top level location for storing CLC data. See section 3.1.2 for further details.
- **Create Folder** ( ). Create new folders within a data location.
- **Update All** ( ). Refresh the view of the Navigation Area.

Between the area listing the data and the icons mentioned above is the Quick Search field, which can be used to find data elements in your CLC Locations. This is described in section 3.4.1.

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.

Elements placed in a folder (e.g. by copy/pasting or dragging) will be put at the bottom of the folder listing. If the element is placed on another element, it will be placed just below that element in the folder listing.

If an element of the same name already exists in that folder, a copy will be created with the name extension "-1", "-2" etc.

Opening and viewing data elements is described in section 2.1.
3.1.1 Data structure

The data in the **Navigation Area** is organized into a number of **Locations**, also known as File Locations or Data Locations. A 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 on the file system that the location points to. The full path to the system folder can be seen by mousing over the data location folder icon (figure 3.2).

![Figure 3.2: Mousing over the 'CLC_Data' location reveals a tooltip showing the full path to the folder on the underlying file system. Here, this is C:\Users\<username>\CLC_Data.](image)

When the **CLC Main Workbench** is started for the first time, there will be a location called **CLC_Data**, which is the default data location.

Adding more locations is described in section 3.1.2. Another location can be specified as the default by right-clicking on the location folder in the Navigation Area and choosing the option **Set as Default Location** from under Location in the menu (figure 3.3). This setting only applies to you. Other people using the same Workbench can set their own default locations.

![Figure 3.3: Data location options are available in a right-click context menu. Here, a new data location is being specified as the default.](image)

Administrators can also change the default data location for all users of a Workbench. This is de-

**Note:** There will also be a location called **CLC_References**. This location is of relevance primarily if you are working with others using a **CLC Genomics Workbench**, who are sharing their results and reference data with you. It is intended for storing genomic references and associated data, downloaded using the Reference Data Manager, distributed with the **CLC Genomics Workbench**.

### Data held on a CLC Server

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.](image)

Locations on a **CLC Server** are configured by the server administrator.

#### 3.1.2 Adding locations

When a Workbench is first installed it has one data location configured by default. Unless your administrator has set things up otherwise, this location will be named **CLC_Data**. It points to the following folder on the underlying system:

- **Windows:** `C:\Users\<your_username>\CLC_Data`
- **Mac:** `~/CLC_Data`
- **Linux:** `/homefolder/CLC_Data`

To add a new location, click on the (folder) icon at the top of the Navigation Area, or go to:

**File | New | Location**

Navigate to the folder you want to add as a data location (see figure 3.5).

The name of the new location will be the name of the folder selected. To see the full path to the folder on the file system, hover the mouse cursor over the location icon (folder).

When you click **Open**, the new location is added to the **Navigation Area** (figure 3.6).
CHAPTER 3. DATA MANAGEMENT AND SEARCH

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

Figure 3.6: The new location has been added. Because this folder had not been used as a CLC location before, index files are built for this area, as recorded in the Processes tab below the Navigation Area.

**Additional notes about CLC locations**

Requirements for folders being used as CLC locations:

- You must have permission to read from that folder, and if you plan to save new data elements or update data elements, you must have permission to write to that folder.
• The folder chosen as a CLC location must not a subfolder of any area already being used as a CLC Workbench or CLC Server location.

Folders on a network drive or a removable drive can act as CLC locations. Please note though that interruptions to file access can lead to problems. For example, if you have set up a CLC location on One Drive, start editing a cloning experiment, and your laptop goes to sleep, unsaved work may be lost, and errors relating to the lost connection may be reported. If your CLC locations are on such systems, enabling offline line access (aka "always available files") can avoid such issues.

Locations appear inactive in the Navigation Area 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 to refresh the view. All available locations will then be shown as active. There can be sometimes be a short delay before the interface update completes.

Sharing data
Sharing data is possible when a network drive is available to multiple Workbenches. In this case, you can add the same folder as a CLC 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 when adding a CLC location already used by other Workbenches as a CLC location, the exact same folder in the file system hierarchy as the other Workbenches have used is the one selected to add as a location. Indicating a folder higher up or lower down in the hierarchy will cause problems with the indexing of the files. This can lead to newly created objects made by Workbench A not being found when searching from Workbench B and vice versa, as well as issues with associations to CLC Metadata Tables.

3.1.3 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 (￼)
- 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.
3.1.4 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.

3.1.5 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.

3.1.6 Moving and copying elements

There are 2 main methods for copying or moving elements using the CLC Workbench:

1. Select the element(s) in the Navigation Area and then use the cut, copy and paste actions. These actions can be carried out using options under the Edit menu:

Cut (✂), Copy (⌘) and Paste (⌘). Alternatively, keyboard shortcuts can be used:

- Ctrl + C (⌘ + C on Mac) to copy
- Ctrl + X (⌘ + X on Mac) to cut
- Ctrl + V (⌘ + V on Mac) to paste

When you cut an element, it will appear "grayed out" until you activate the paste function. You can revert the cut command by copying another element.

2. Drag and drop elements:

- Move a data element between folders in the same File Location by dragging the element to the new folder.
- Make a copy of the element in in a different File Location by dragging the element to the new location.
- Make a copy of a data element in the same File Location as the original by keeping the Ctrl / Command key depressed while dragging the data element to the new folder.

File Locations are described in section 3.1.2.
If an element of the same name already exists in the destination folder, the name of the copied/moved element will get a numeric extension, ",-1", ",-2", etc.

If you copy an element, and then paste into a place that expects text, a CLC URL for the location of that element will be pasted in. This can be a convenient way to share the location of specific elements with colleagues if they are using shared data locations. For example, you could paste the CLC URL into an email or chat message. To find an element, paste the URL into the Search field above the Navigation Area in a CLC Workbench and press Enter.

Dragging and dropping elements can also be used to:

- **Open** an element in the View Area by clicking on its name in the Navigation Area and dragging it to the View Area.
- **Save** open elements by clicking on its tab in the View Area and dragging that tab to the desired location in the Navigation Area.
- **Move tabs in the View Area** Open elements in the View area can be moved around by dragging their tabs. This is described further in section 2.1.4.

### 3.1.7 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 [12.3.3](#).

### 3.1.8 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, using the **Delete** (\(\text{Delete}\)) option from the **Edit** menu, the right-click menu of an element, or in the **Toolbar**, or by just using the **Delete key** of your keyboard.

This will cause the element to be moved to the Recycle Bin (\(\text{Recycle Bin}\)) 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 [12.3.5](#).

**Items in a recycle bin can be restored** in two ways: by dragging the elements with the mouse into the folder where they used to be, or by right-clicking on the element and choosing the option **Restore from Recycle Bin**. 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 using the **Empty Recycle Bin** command available under the **Edit** menu or in the menu presented when you right-click on a Recycle Bin (\(\text{Recycle Bin}\)). This deletes the data and frees up disk space.

**Note!** Emptying the recycle bin cannot be undone. Data is not recoverable after it has been 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 (\(\text{Recycle bin}\)) 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.
3.1.9 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.

![Figure 3.7: Viewing the elements in a folder.](image)

When the elements are shown in the view, they can be sorted by clicking the heading of each of the columns. You can further refine the sorting by pressing Ctrl ( on Mac) while clicking the heading of another column.

Sorting the elements in a view does not affect the ordering of the elements in the Navigation Area.

**Note!** The view only displays one "layer" at a time: the content of subfolders is not visible in this view. Also note that only sequences have the full span of information like organism etc.

**Batch edit folder elements** You can select a number of elements in the table, right-click and choose Edit to batch edit the elements. In this way, you can change for example the description or name of several elements in one go.

In figure 3.8 you can see an example where the name of two sequence are renamed in one go. In this example, a dialog with a text field will be shown, letting you enter a new name for these two sequences.

**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 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 3.9 shows an example of a typical table. This is the table result of Find Open Reading Frames (OCR). We use this table as an example to illustrate concepts relevant to all kinds of tables.

**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 or unchecking the boxes beside column names in 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 types (such as Expression Browsers), the content of some columns can be modified using settings in the Side Panel (such as Expression values and Grouping in figure 3.10).

![Figure 3.10: A table showing the results of an RNA-Seq analysis.](image)

**Working with tables using the right-click menu**

Right-clicking on a table reveals standard menu options, as well as table-specific options, found under these submenus:

- **Table filters** Lists advanced filters relevant to the column that you right-click upon. See section 3.2.1 for details.
- **File** Includes the option Export Table, which allows the table to be exported in .csv or Excel format. This option respects the filtering, sorting and column selections that have been applied to the table.
- **Edit** Includes the option Copy Cell, which copies the contents of individual cells to the clipboard.

**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.

**Adjusting the column order** The column order for viewing and exporting purposes can be changed. To move a column, click on its heading and, keeping the mouse button depressed, drag it to the desired location in the table. Files exported from a table at this point, such as .csv files, will reflect the new column order.

It is not possible to save the revised column order in the Workbench.
3.2.1 Filtering tables

Filters can be set using the functionalities located at the top of any table in the Workbench: a Filter to Selection button, a simple filter mode and an advanced filter mode. A counter in the upper left corner tells you the number of rows that passed the filter.

Filter to selection  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.

Simple filter  The simple mode is the default and is applied simply by typing text or numbers (see an example in figure 3.11).

![Figure 3.11: Typing "neg" in the filter in simple mode.](image)

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.

The following characters have special meanings when used in simple filtering of tables in the workbench:

- **Space** (separates search items unless inside quotes)
- **Backslash** (escapes characters, in particular those mentioned in this list)
- **Single and double quotes ' and "** (define entire phrases to search for)
- **Minus -** (specifies words or phrases to exclude)
- **Colon :** (searches in a specific table column)

These characters cannot be used in the Advanced filter described below, because the Advanced Filter functionality makes it easy to include/exclude specific terms or limit searches to a particular column by providing the appropriate fields.

Also note that typing *cat dog* in the Simple filter field will return all rows with cat and dog in them, in any order. But the same search term put in the Advanced filter field (visible once you click on the little arrow to the right of the simple filter field), will only return rows with the exact phrase "cat dog".
Advanced filter  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.

- = (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 3.12 shows an example of an advanced filter which displays the open reading frames larger than 400 that are placed on the negative strand.

Figure 3.12: The advanced filter showing open reading frames larger than 400 that are placed on the negative strand.

3.3 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 available\(^1\) go to the Workbench:

right-click the data location | Location | Attribute Manager

This will display the dialog shown in figure 3.13.

---

\(^1\)If the data location is a server location, you need to be a server administrator to do this.
Click the **Add Attribute** (➕) button to create a new attribute. This will display the dialog shown in figure 3.14.

![Create Attribute dialog]

**Figure 3.14: 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).

- **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 a data element is copied, attribute values are transferred to the copy of the element by default. To prevent the values for an attribute from being copied, uncheck the **Values are inheritable** checkbox.

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 3.15).

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.

### 3.3.1 Filling in values

When a set of attributes has been created (as shown in figure 3.16), the end users can start filling in information.

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 3.17.

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.

---

*Figure 3.15: Defining items in a list.*
When nobody has entered information, the attribute will have a "Not set" written in red next to the attribute (see figure 3.18).

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 3.18, 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 **3.1.9**).

### 3.3.2 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.

### 3.3.3 Searching custom attributes

When a text-based attribute is created, it becomes available for searching using Quick Search or Local Search.

**Quick Search:** Precede the name of the attribute with "attrib_" and then add a colon followed by the query term, e.g. "attrib_color:blue".

You must use whole words when searching for elements containing particular values for local attributes.

For searching purposes, **words** are the terms on either side of a space, hyphen or underscore in a name. The names of elements and folders are split into words when indexing.

In the **Local Search** tool, the local attributes are available to select from a drop down list, in addition to Name and Path.


### 3.4 Searching for data in CLC Locations

There are two ways of searching for data in your CLC Locations:

- **Quick Search** Available above the **Navigation Area** and described in section **3.4.1**. By default, terms entered are used to search for the names of data elements and folders across all available CLC Locations.

- **Local Search** Available under the Utilities menu and described in section **3.4.2**. All CLC Locations can be searched, or an individual Location can be searched. Local searches can be saved, so that the same search can be run again easily.
The CLC Main Workbench automatically maintains an index of data in each CLC Location. These indexes are used for searches.

Problems with search results can reflect a problem with an index. If a search does not return the results you expect, re-building the index may help. Do this by right-clicking on the relevant data location in the Navigation Area and then selecting:

Location | Rebuild Index

Rebuilding the index for locations with a lot of data can take some time.

The index building process can be stopped under the Processes tab, described in section 2.4.

Indexes are updated automatically when data is moved between CLC Locations, but not when data elements are moved within a given CLC Location. When searching based on names, this does not matter. However, for searches based on information in the path, the index may need to be rebuilt before searching.

### 3.4.1 Quick Search

The search field just above the Navigation Area on the left side of the CLC Main Workbench (figure 3.19) can be used to search for data elements and folders in all available CLC Locations. By default matches against names are reported. Searches are case insensitive.

A CLC URL, which specifies a specific element or folder, can also be entered. This is described at the end of this section.

**Using Quick Search for searches based on names**

The examples below refer to searches where the following elements are present in a CLC Location.

- E. coli reference sequence
- Broccoli sequence
- Coliform set

**Search with a single term** to look for any element or folder with a name containing that term.

*Example 1:* A search for coli would return all 3 elements listed above.

**Search with two or more terms** to look for any element or folder with a name containing all of those terms.

*Example 2:* A search for coli set would return "Coliform set" but not the other two entries listed in the earlier example.

**Search with two or more words in quotes** to look for any element or folder name containing those words, appearing consecutively, in the order provided. Whole words must be used within quotes, rather than partial terms.

For searching purposes, **words** are the terms on either side of a space, hyphen or underscore in a name. The names of elements and folders are split into words when indexing.
Example 3: A search for "coli reference" would find an element called "E. coli reference sequence".

Example 4: A search for "coli sequence" would not return any of the elements in the example list. In the name "E. coli reference sequence", the words coli and sequence are not placed consecutively, and in "Broccoli sequence", "coli" is a partial term rather than a whole word.

Why only words when searching with quotes? The use of quotes allows quite specific searches to be run quickly, but only using words, as defined by the indexing system.

Tip: Searches with whole words are faster than searching with partial terms. If a term is a word in some names but a partial term in others, the hits found using the complete word are returned first. E.g. searches with the term cancer would return elements with names like "cancer reads" and "my cancer sample" before an element with a names like "cancerreads".

Note: Wildcards (*, ?, ~) are ignored in basic searches. If you wish to define a search using wildcards, use the advanced search functionality of Quick Search.

Advanced searching using Quick Search

When a search term is preceded by "name: ", more specific searches can be defined, using standard wild cards. If multiple query terms are provided, elements and folders containing all terms are returned. Searches are case insensitive.

Terms in the path can also be searched for by preceding the term with "path: ".

If you have defined local (custom) attributes for any of your CLC Locations, the contents of these can also be searched with using the name of the attribute preceded by "attrib ", e.g. "attrib_color:blue".

Example 5: A search for path:tutorials name:cancer would find all data elements or folders where at least one folder in its path contained "tutorials" and the name of the folder or element included the word "cancer". Word here is as defined earlier in this section. In this example, an element named "cancer tissue reads" in a subfolder under "CLC_Tutorials" would be found, but an element called "cancerreads" in that subfolder would not be.

Example 6: A search for path:tutorials cancer would find all data elements or folders where at least one folder in its path contained "tutorials" (with capital or small letters) and the name of the folder or element reported including the word "cancer". In this example, an element named "cancer tissue reads" in a subfolder under "CLC_Tutorials" would be found, and so would an element called "cancerreads" in that subfolder.

Figure 3.19: *Enter terms in the Quick Search field to look for elements and folders.

Advanced search expressions

The terms below can only be used when the field being searched has been specified explicitly, e.g. by preceding the search term with "name:" or "path: ".

- **Wildcard multiple character search (**). Appending an asterisk * to the search term find
matches that start with that term. E.g. A search for \texttt{BRCA*} will find terms like \textit{BRCA1}, \textit{BRCA2}, and \textit{BRCA1gene}.

- **Wildcard single character search (?).** The ? character represents exactly one character. For example, searching for \texttt{BRCA?} would find \textit{BRCA1} and \textit{BRCA2}, but would not find \textit{BRCA1gene}.

- **Search related words (~).** Appending a tilde to the search term looks for fuzzy matches, that is, terms that almost match the search term, but are not necessarily exact matches. For example, \texttt{ADRA1~} will find terms similar to \textit{ADRA1A}.

**Search results**

When there are many hits, only the first 50 are shown initially. To see the next 50, click on the Next (\rightarrow) arrow just under the list of results.

The number to return initially can be configured in the Workbench Preferences, as described in section 4.

**Finding the element or folder in the Navigation Area**

You can move the focus from a search result to its location in the Navigation Area by doing one of the following:

- Click on a search result while keeping the Alt key depressed.
- Right-click on a search result and click on Show Location in the menu presented.

**Quick Search history**

You can access the 10 most recent searches by clicking the icon (Q) to the left of the search field (figure 3.20). When one of these is selected, the search is automatically run again.

**Searching using CLC URLs**

A particular data element or folder can be quickly found by entering its CLC URL into the Quick Search box (figure 3.21). An example of when this is useful is when working using a CLC Server and sharing the location of particular data elements with another person.

To obtain a CLC URL for an element or folder, right click on it in the Navigation Area and choose the option Copy. Alternatively, select it in the Navigation Area and copy it using the keyboard shortcut Ctrl + C.

When you paste into the Quick Search box, it is the CLC URL that is pasted there. CLC URLs can also be pasted anywhere where text is expected, e.g. an email or text file.

**3.4.2 Local Search**

The Local Search tool is useful when searching large data locations and if you wish to save searches to be run multiple times.
CHAPTER 3. DATA MANAGEMENT AND SEARCH

Figure 3.20: Recent searches are listed and can be selected to be re-run by clicking on the icon to the left of the search field.

Figure 3.21: Data elements can be located using a CLC URL.

To launch the Local Search tool, go to:

Utilities | Local Search ( )

or Ctrl + Shift + F ( + Shift + F on Mac)

You can choose to search across all CLC Locations or you can specify a single Location to search.

You can search for terms in the names of elements or folders, as well as terms in the path to those elements or folders. If you have defined local (custom) attributes for any of your CLC Locations, the contents of these can also be searched.

More than one search term can be added by clicking on the Add search parameters button. To search for terms of the same type (e.g. terms in names), you can just add multiple terms in the same search field, as described below.

Click on the Search button to start the search.

Using Local Search for searches based on names

The examples below refer to searches where the following elements are present in a CLC Location.

- E. coli reference sequence
- Broccoli sequence
- Coliform set
Search with a single term to look for any element or folder with a name containing that term.

Example 1: A search for *coli* would return all 3 elements listed above.

Search with two or more terms to look for any element or folder with a name containing all of those terms.

Example 2: A search for *coli set* would return "Coliform set" but not the other two entries listed in the earlier example.

Search with two or more words in quotes to look for any element or folder name containing those words, appearing consecutively, in the order provided. Whole words must be used within quotes, rather than partial terms.

For searching purposes, words are the terms on either side of a space, hyphen or underscore in a name. The names of elements and folders are split into words when indexing.

Example 3: A search for "coli reference" would find an element called "E. coli reference sequence".

Example 4: A search for "coli sequence" would not return any of the elements in the example list. In the name "E. coli reference sequence", the words coli and sequence are not placed consecutively, and in "Broccoli sequence", "coli" is a partial term rather than a whole word.

Why only words when searching with quotes? The use of quotes allows quite specific searches to be run quickly, but only using words, as defined by the indexing system.

Tip: Searches with whole words are faster than searching with partial terms. If a term is a word in some names but a partial term in others, the hits found using the complete word are returned first. E.g. searches with the term cancer would return elements with names like "cancer reads" and "my cancer sample" before an element with a names like "cancerreads".

Note: Wildcards (*, ?, ~) are ignored in basic searches. If you wish to define a search using wildcards, use the advanced search functionality of Quick Search.

Saving Local Searches

Local searches can be saved ( ), so that the same search can be run again easily. Saving a search saves the query, not the results.

The search you just set up can be saved by doing one of the following:

- Choose Save As... from under the File menu, or
- Click on the tab of the search view and drag and drop it into a folder in the Navigation Area.

These actions save the search query. (It does not save the search results.)

This can be useful when you run the same searches periodically.
Chapter 4

User preferences and settings

Contents

4.1 General preferences ................................................. 86
4.2 View preferences .................................................. 87
   4.2.1 Import and export Side Panel settings ...................... 89
4.3 Data preferences .................................................. 90
4.4 Advanced preferences ............................................. 90
4.5 Export/import of preferences .................................... 92
4.6 Side Panel view settings ......................................... 93

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 (\textcopyright)

or

   Ctrl + K (\textasciicircum \textasciicircum + ; on Mac)
CHAPTER 4. USER PREFERENCES AND SETTINGS

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 2.1.3).

- **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 that no matter whether **Audit Support** is checked or not, all changes are also recorded in the History log (see section 2.5).

- **Number of hits.** The number of hits shown in **CLC Main Workbench**, when e.g. searching NCBI. (The sequences shown in the program are not downloaded, until they are opened or dragged/saved into the Navigation Area).

- **Locale Setting.** Specify which country you are located in. This determines how punctuation is used in numbers all over the program.

- **Show Dialogs.** A lot of information dialogs have a checkbox: “Never show this dialog again”. When you see a dialog and check this box in the dialog, the dialog will not be shown again.
If you regret and wish to have the dialog displayed again, click the button in the General Preferences: Show Dialogs. Then all the dialogs will be shown again.

- **Usage information.** When this item is checked, anonymous information is shared with QIAGEN about how the Workbench is used. This option is enabled by default.

  The information shared with QIAGEN is:
  
  - Launch information (operating system, product, version, and memory available)
  - The names of the tools and workflows launched (but not the parameters or the data used)
  - 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.](image)

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.

![Figure 4.5: Number formatting of tables.](image)

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).
   - 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.](image)

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 13.2).
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.](image)

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.](image)

WARNING! If you choose to overwrite the existing settings, you will loose 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 7.1).
- **Graphics** export of the views which creates image files in various formats (described in section 7.3).
- Import and export of **Side Panel Settings** as described above.
- Import and export of all the **Preferences** except the Side Panel settings. This is described in the previous section.

### 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 21.5.1).

### 4.4 Advanced preferences

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

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

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

![Figure 4.9: Adjusting proxy settings.](image-url)
List hosts that should be contacted directly, i.e. not via the proxy server, in the **Exclude hosts** field. The value can be a list, with each host separated by a `|` symbol. The wildcard character `*` can also be used. For example: `*.foo.com|localhost`.

The proxy can be bypassed when connecting to a **CLC Server**, as described in section 6.1.

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

**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 3.1.2). 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. 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.

Compatibility information:

- A new compression method was introduced with version 22.0 of the CLC Genomics Workbench, CLC Main Workbench and CLC Genomics Server. Compressed data created using those versions can be read by version 21.0.5 and above, but not earlier 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. Compressed data created using these versions is not compatible with older versions of the software. Data created using these versions can be opened by later versions of the software, including versions 22.0 and above.

To share specific data sets for use with software versions that do not support the compression applied by default, we recommend exporting the data to CLC or zip format and turning on the export option "Maximize compatibility with older CLC products". See section 7.2.4.

**NCBI Integration**  Without an API key, access to NCBI from asingle IP-address is limited to 3 requests per second; if many workbenches use the same IP address when running the 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](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 4.10).

![Figure 4.10: Select which of the preferences you want to export.](image)

**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 7.1).

- **Graphics** export of the views that create image files in various formats (described in section 7.3).

- Import and export of **Side Panel Settings** as described in the next section.

- Import and export of all the **Preferences** except the Side Panel settings. This is described above.
4.6 Side Panel view settings

The Side Panel, on the right side of open views, contains settings for customizing a view. The settings available are specific to the view type. For example, the Side Panel settings for the graphical view of a sequence differ from those for the Annotation Table view. This section describes how these settings can be saved, applied and shared.

For general information about Side Panels, see section 2.1.6.

Saving the settings for a view

After adjusting the settings in the Side Panel, click on the *Save View...* button, located at the bottom on the right. (figure 4.11). This launches a dialog that can be used for saving, applying, removing and exporting/importing view settings (figure 4.12).

The top part of this dialog is used when saving view settings.

![Figure 4.11: Functionality in the bottom of the Side Panel. Click on the Save View button to save the current settings in the Side Panel for future use.](image)

Views can be saved to be used by:

- **Just the element you are working on.** Enter a new name for this view setting, leave the `Save for all <element type> views` box unchecked and click on the *Save* button.
  These settings will then be used by default whenever you open up this element to the relevant view. If this element is exported to CLC format, these view settings are included. These settings will not be available for use by other elements.

- **Any element with that view** Enter a new name for this view setting, check the `Save for all <element type> views` box and click on the *Save* button.
  This view setting will now be available for elements with the relevant view type. For example, if your settings were for Table view, the settings will be available to be applied to other elements with the Table view.

View settings are user-specific, so if others also use this Workbench, they will not see these settings you have configured. Sharing view settings is described below.

Applying saved settings

Click on the *Save View...* button (figure 4.11) to launch the dialog for saving, applying, removing and exporting/importing view settings (figure 4.12).

Apply saved view settings using the functionality in the second part of the dialog.

Views can be applied to:
• **Just the element you are working on.** Select from a setting from the list of existing view settings, leave the **Use as a standard view setting for <element type> view** box unchecked and click on the **Apply** button.

These settings will then be used by default whenever you open up this element to the relevant view.

• **All elements with that view** Select from a setting from the list of existing view settings, check the **Use as a standard view setting for <element type> view** box and click on the **Apply** button.

This view setting will now be applied to all elements with the relevant view type. For example, if the settings were for Table view, the settings will be applied to other elements with the Table view.

Settings applied this way are not part of the element itself, and thus are not included with the element if it is exported.

The CLC Standard Settings option applies the default options for the view.

![Save View Settings](image)

**Figure 4.12: Functionality available at the bottom of Side Panels.**

**Removing saved view settings**

Click on the **Save View...** button (figure 4.11) to launch the dialog for saving, applying, removing and exporting/importing view settings (figure 4.12).

View settings can be removed using the functionality in the third part of the dialog.

Select the name of the view setting to remove from the drop-down menu and click on the **Remove** button.

**Sharing view settings**

Saved view settings can be exported, and then shared with others using CLC Workbenches.
Click on the **Save View...** button (figure 4.11) to launch the dialog for saving, applying, removing and exporting/importing view settings (figure 4.12). The relevant options are near the bottom of the dialog.

Click on the **Export...** button, and in the resulting window select the view settings to be exported. Click on the **Export...** button in that window and then provide a filename. A file of that name with the suffix `.vsf` appended will be saved to the location you indicate.

View settings files (.vsf) can be imported by clicking on the **Import...** button.

Exporting and importing saved settings can also be done in the **Preferences** dialog under the **View** tab (see section 4.2.1).
Chapter 5

Printing

Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Selecting which part of the view to print</td>
<td>97</td>
</tr>
<tr>
<td>5.2 Page setup</td>
<td>98</td>
</tr>
<tr>
<td>5.3 Print preview</td>
<td>100</td>
</tr>
</tbody>
</table>

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 7.3) in a graphic format, and then import it into a document or a presentation.

All the kinds of data that you can view in the View Area can be printed. The CLC 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 (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.
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.

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.
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.
• **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 5.5: Page Setup.](image)

![Figure 5.6: An example where Fit to pages horizontally is set to 2, and Fit to pages vertically is set to 3.](image)

**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.

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

Connections to other systems

Contents

<table>
<thead>
<tr>
<th>6.1 CLC Server connection</th>
<th>6.2 AWS Connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>..........................</td>
<td>..........................</td>
</tr>
<tr>
<td>........................................</td>
<td>101 ..........................</td>
</tr>
<tr>
<td>........................................</td>
<td>103 ..........................</td>
</tr>
</tbody>
</table>

Under the Connections menu are tools for connecting the CLC Main Workbench to other systems.

6.1 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.
- Workflows installed on the CLC Server will be available to launch from the Toolbox.
- External applications configured and enabled on the CLC Server will be available to launch from the Toolbox, and to include in workflows.

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

File | CLC Server Connection

This will bring up a login dialog as shown in figure 6.1.

Your server administrator should be able to provide you with the necessary details to fill in the fields. 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 9.1.1.
- Monitoring processes sent to the CLC Server from a CLC Workbench is described in section 2.4.
- 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 3.1.8.
- Importing data to a CLC Server and exporting data held on a CLC Server is described in section 7.5.

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 at [http://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=User_authentication_using_Workbench.html](http://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=User_authentication_using_Workbench.html).

**Bypassing proxy server settings**

After proxy settings have been configured (section 4.4), and the CLC Main Workbench has been restarted, a checkbox is shown in the CLC Server Connection dialog allowing the proxy settings to be bypassed for connections to the CLC Server (figure 6.2).

If you wish to preconfigure Workbenches to bypass the proxy settings when connecting to a CLC Server, this can be done using a proxy.properties file, where the IP address (not the host name) of the CLC Server is provided in the **proxyexclude** field. For further information on using properties files, see [https://resources.qiagenbioinformatics.com/manuals/workbenchdeployment/current/index.php?manual=Per_computer_Workbench_information.html](https://resources.qiagenbioinformatics.com/manuals/workbenchdeployment/current/index.php?manual=Per_computer_Workbench_information.html).
6.2 AWS Connections

AWS connections are used when:

- Accessing AWS S3 locations, to import data from or export data to.
- Submitting analyses to a CLC Genomics Cloud setup, if available on that AWS account.

Working with data on S3 via the Workbench is of particular relevance when submitting jobs to run on a CLC Genomics Cloud setup making use of functionality provided by the CLC Cloud Module.

When launching workflows to run locally using on-the-fly import and selecting files from AWS S3, the files selected are first downloaded to a temporary folder and are subsequently imported.

All traffic to and from AWS is encrypted using a minimum of TLS version 1.2.

Configuring AWS credentials

To configure an AWS connection, go to:

Connections | AWS Connections

After a connection has been configured, it will look like that in figure 6.3, where AWS connections are listed, along with information about their status. These can be edited or removed, if desired. The status is indicated using colors. Green indicates the connection is valid and ready for use.

Connections to a CLC Genomics Cloud are indicated in the CGC column. To submit analyses to the CLC Genomics Cloud, the CLC Cloud Module must be installed and a license for that module must be available.

Click on the Add AWS Connection button to configure an AWS connection. Enter the following information (figure 6.4):
Figure 6.3: The configuration dialog for AWS connections. A valid connection has been configured and S3 locations will be available via exporters and relevant importers in the Workbench.

- **Connection name**: A short name of your choice, identifying the AWS account. This name will be shown as the name of the data location when importing data to or exporting data from Amazon S3.
- **Description**: A description of the AWS account (optional).
- **AWS access key ID**: The access key ID for programmatic access for your AWS IAM user.
- **AWS secret access key**: The secret access key for programmatic access for your AWS IAM user.
- **AWS region**: An AWS region. Select from the drop-down list.
- **AWS partition**: The AWS partition for your account.

The dialog continuously validates the settings entered. When they are valid, the Status box will contain the text "Valid" and a green icon will be shown. Click on **OK** to save the settings.

Figure 6.4: Configuration of an AWS Connection in a CLC Workbench

AWS credentials entered are stored, obfuscated, in Workbench user configuration files.

**Exporting data to AWS S3**

To export data to an AWS S3 location, launch the exporter, and when prompted for an export location, select the relevant option from the drop-down menu (figure 6.5).
Figure 6.5: After an AWS connection is selected when exporting, you can select the S3 bucket and location within that bucket to export to.
Chapter 7

Import/export of data and graphics

Contents

7.1 Standard import .................................................. 106
    7.1.1 External files ............................................. 107

7.2 Data export ..................................................... 108
    7.2.1 Export formats .......................................... 108
    7.2.2 Export parameters ....................................... 110
    7.2.3 Specifying the exported file name(s) .................. 111
    7.2.4 Export of folders and data elements in CLC format .... 112
    7.2.5 Export of dependent elements ......................... 114
    7.2.6 Export of tables ......................................... 114
    7.2.7 GFF3 export .............................................. 116
    7.2.8 JSON export .............................................. 116
    7.2.9 Graphics export ......................................... 119
    7.2.10 Export history .......................................... 121
    7.2.11 Backing up data from the CLC Workbench ............. 122

7.3 Export graphics to files ......................................... 123
    7.3.1 File formats .............................................. 126

7.4 Export graph data points to a file .............................. 128

7.5 CLC Server data import and export ............................ 129

7.6 Copy/paste view output ........................................ 130

A large number of data formats can be imported into the CLC Main Workbench. Data types that are not recognized are imported as "external files". Such files are opened in the default application for that file type on your computer (e.g. Word documents will open in Word).

This chapter describes importing and exporting data in bioinformatic data formats and as external files, as well as how to export graph data points to a file, and how to export graphics.

7.1 Standard import

CLC Main Workbench has support for a wide range of bioinformatic data formats. See section H.1.1 for a list.
The Standard Import tool  To import data, click on the Import icon in the Toolbar and choose Standard Import.

Alternatively, go to File | Import | Standard Import.

Select files to import by clicking on the Add files button. Alternatively, specify folders containing files to import by clicking on the Add folders button (figure 7.1).

The default option is Automatic import. The file format is automatically detected based on a combination of the file extension (e.g. .fa for fasta) and detection of file contents specific to particular formats. Based on this, the relevant importer is run. The particular importer used is recorded in the element history. If the format is not supported, the file is imported as an external file.

Note: If you drag and drop files into the Navigation Area, standard import is run with automatic detection of the file format.

You can specify the format explicitly using the option Force import as type.

The Force import as external file option can be useful if you are trying to import a standard format file, such as a text file, but it is being detected as bioinformatics format file, such as sequence data.

Figure 7.1: The Standard Import tool.

Importing by copy/pasting text  Data can be copied and pasted directly into the Navigation Area.

Copy text | Select a folder in the Navigation Area | Paste

Standard import with automatic format detection is run using the pasted content as input.

This is a fast way to import data, but importing files as described above is less error prone, and thus generally recommended instead.

7.1.1 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
CHAPTER 7. IMPORT/EXPORT OF DATA AND GRAPHICS

by double-clicking the file in the Navigation Area. The file is opened using the default application for this file type (e.g. Microsoft Word for .doc-files and Adobe Reader for .pdf).

External files are imported and exported in the same way as bioinformatics files (see section 7.1). 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).

7.2 Data export

Data can be exported from the CLC Main Workbench to many standard formats. Supported formats are listed in section H.1.1, 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 (Ctrl+E)

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

File | Export with Dependent Elements

This tool is described further in section 7.2.5.

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 Finish.

7.2.1 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 (figure 7.2).

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 7.3. 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.

- 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. See section 7.2.4.

![Figure 7.2: 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.](image)

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.
CHAPTER 7. IMPORT/EXPORT OF DATA AND GRAPHICS

Selecting data for export In figure 7.4 we show the selection of a variant track for export to VCF format.

Further information is available about exporting the following types of information:

- Tables: section 7.2.6
- Reports to JSON format: section 7.2.8
- Graphics to a range of formats: section 7.2.9
- Data element history: section 7.2.10

### 7.2.2 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 7.5.

Examples of configuration options:

- **Maximize compatibility with older CLC products** This is described in section 7.2.4.
- **Compression options** Within the **Basic export parameters** section, you can choose to compress the exported files. The options are no compression (None), gzip or zip format. Choosing zip format results in all data files being compressed into a single file. Choosing gzip compresses the exported file for each data element individually.
Figure 7.5: Configure the export parameters. When exporting to CLC format, you can choose to maximize compatibility with older CLC products.

- **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.

### 7.2.3 Specifying the exported file name(s)

The names to give exported files can be configured in the export wizard. Names can be specified directly or placeholders can be used. Placeholders specify particular types of information, and thus are a convenient way to apply a consistent naming pattern to many exports.

The default is to use the placeholders `{name}` and `{extension}`, as shown in figure 7.6. Using these, the original data element name is used as the basename of the exported file, and the file format is used as the suffix. The actual filename that would result is shown in the Output file name field for the first element being exported.

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

An example where specific text instead of a placeholder might be preferred would be if the extension used for a particular format is not as desired. For example, the extension used for fasta files is `.fa`. To use `.fasta` instead, replace `{extension}` with `".fasta` in the Custom file name field, as shown in figure 7.7.

When exporting a single file, the desired filename can just be typed in the Custom file name field. This should not be done when exporting to more than one file, as this would result in every exported file having an identical name.
The following placeholders are available:

- `{name}` or `{1}` - default name of the data element being exported
- `{extension}` - default extension for the chosen export format
- `{counter}` - 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.
- `{host}` - name of the machine the job is run on
- `{user}` - name of the user who launched the job
- `{year}`, `{month}`, `{day}`, `{hour}`, `{minute}`, and `{second}` - timestamp information based on the time an output is created. Using these placeholders, items generated by a tool at different times can have different filenames.

**Note:** Placeholders available for Workflow Export elements are different and are described in section 11.2.3.

Exported files can be saved into subfolders by using a forward slash character `/` at the start of the custom file name definition. When defining subfolders, all later forward slash characters in the configuration, except the last one, are interpreted as further levels of subfolders. For example, a name like `/outputseqs/level2/myoutput.fa` would put a file called `myoutput.fa` into a folder called `level2` within a folder called `outputseqs`, which would be placed within the output folder selected in the final wizard step when launching the export tool. If the folders specified in the configuration do not already exist, they are created. Folder names can also be specified using placeholders.

### 7.2.4 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 can export to CLC format, and they can run the CLC Main Workbench in Viewing Mode, which does not require a license (section 1.4.7). In Viewing Mode, CLC format data can be imported and viewed in the same way it would be using a licensed Workbench.

**Compatibility of the CLC data format between Workbench versions**

When exporting to CLC or zip format, the option "Maximize compatibility with older CLC products" is presented at the Specify export parameters step. With that option checked, data is exported without internal compression applied. This is only relevant when sharing data with someone using a version of the CLC software where the compression applied by default is not supported.

Compatibility information:

- A new compression method was introduced with version 22.0 of the CLC Genomics
CHAPTER 7. IMPORT/EXPORT OF DATA AND GRAPHICS

Workbench, CLC Main Workbench and CLC Genomics Server. Compressed data created using those versions can be read by version 21.0.5 and above, but not earlier 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. Compressed data created using these versions is not compatible with older versions of the software. Data created using these versions can be opened by later versions of the software, including versions 22.0 and above.

Information on how to turn off internal data compression entirely is provided in section 4.4. We generally recommend, however, that data compression remains enabled.

7.2.5 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 (Ctrl I) | 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 7.2.4.

7.2.6 Export of tables

Data in tables can be exported to CSV, tab-delimited, Excel, or HTML format files.

The "Export all columns" option is selected by default. When it is deselected, options for selecting the columns to export are presented in the next wizard step.
When selecting specific columns for export, the option "Export the table as currently shown" is particularly useful if you have filtered, sorted, or selected particular columns in a table that is open in a View. In this case, the effects of these manipulations are preserved in the exported file. This option is not available for all data types.

When the "Export the table as currently shown" is unchecked or disabled, checkboxes for each column to be exported are available to select or deselect. The buttons below that section can help speed up the process of column selection:

- **All** Select all possible columns.
- **None** Clear the existing selection.
- **Default** Select a standard set of columns, as defined by the software for this data type.
- **Last export** Select the columns that were selected during the most recent previous export.
- **Active View** Select the same set of columns as those selected in the Side Panel of the open data element. This button is only visible if the element being exported is in an open View.

In the final wizard step, select the location where the exported elements should be saved.

![Selecting table columns to be exported.](image)

The data exported will reflect any filtering and sorting applied.

**Considerations when exporting tables**
• **Row limits** 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 subset your data and then export it to several worksheets, where each would have fewer than 66,530 rows.

• **Decimal places** When exporting to CSV, tab-separated, or Excel formats, numbers with many decimals are exported with 10 decimal places, or in scientific notation (e.g. 1.123E-5) when the number is close to zero.

When exporting a table in HTML format, data are exported with the number of decimals that have been defined in the *CLC Main Workbench* preference settings. When tables are exported in HTML format from a *CLC Server* the default number of decimal places is 3.

• **Decimal notation** When exporting to CSV and tab delimited files, decimal numbers are formatted according to the Locale setting of the *CLC Main Workbench* (see General preferences 4.1). If you open the CSV or tab delimited file with software like Excel, that software and the *CLC Workbench* should be configured with the same Locale.

### 7.2.7 GFF3 export

All the annotations on sequences (individual elements or in sequence lists) can be exported to a GFF3 format file using the dedicated GFF3 exporter.

A subset of annotations on a sequence or sequence list can be exported from the Annotation Table (**) view: select the annotations to export in the table, right-click, and choose **Export Selection to GFF3 File** from the menu (figure 12.20). Further information about sequence annotations is in section 12.3.

### 7.2.8 JSON export

Reports can be exported in JSON format. An exported JSON file contains 4 main elements:

• **header.** Contains information about the version of the JSON exporter and front page elements included in the report (the front page elements are visible in the PDF export of the report).

• **data.** Contains the actual data found in the report (sections, subsections, figures, tables, text).

• **metadata.** Contains information about metadata files the report referenced to.

• **history.** Contains information about the history of the report (as seen in the "Show history" view).

The data section contains nested elements following the structure of the report:

• The keys of sections (and subsections, etc) are formed from the section (and subsection, etc) title, with special characters replaced. For example, the section "Counted fragment by type (total)" is exported to an element with the key "counted_fragments_by_type_total".

• A section is made of the section title, the section number, and all other elements that are nested in it (e.g., other subsections, figures, tables, text).
Figures, tables and text are exported to elements with keys "figure_n", "table_n" and "text_n", n being the number of the elements of that type in the report.

Figures contain information about the titles of the figure, x axis, and y axis, as well as legend and data. This data is originally available in the Workbench by double clicking on a figure in a report and using the "Show Table" view.

The names of table columns are transformed to keys in a similar way to section titles.

Once exported, the JSON file can be parsed and further processed. For example, using R and the package jsonlite, reports from different samples can be jointly analyzed. This enables easy comparison of any information present in the original reports across samples.

Example of R script to generate a combined RNA-Seq report

R scripts can be used to process reports in JSON format. The script below is an example of how RNA-Seq reports generated by the CLC Genomics Workbench could be processed to compare information across samples. The same idea could be used for processing other sorts of reports.

The script uses jsonlite to parse all the JSON reports. Plots are produced using the package ggplot2. This script is intended for inspiration only and is not supported.

```r
library(jsonlite)
library(tools)
library(ggplot2)

The script relies on the following functions to extract the data from the parsed JSON files.

```
STATS <- c(total_reads, STATS)
names(STATS) <- c("reads_count", "single_mapped", "single_not_mapped",
"paired_mapped_pairs", "paired_broken_pairs",
"paired_not_mapped")
return(data.frame(sample = basename(file_path_sans_ext(report)),
  t(STATS)))

# Get the paired distance from a parsed report. Returns null if the reads were
# unpaired.
get_paired_distance <- function(parsed_report) {
  section <- parsed_report$data$read_quality_control
  if (!("paired_distance" %in% names(section))) {
    return(NULL)
  } else {
    figure <- section$paired_distance$figure_1
    return(data.frame(sample = basename(file_path_sans_ext(report)),
                       figure$data))
  }
}

# Get the figure, x axis, and y axis titles from the paired distance figure
# from a parsed report. Returns null if the reads were unpaired.
get_paired_distance_titles <- function(parsed_report) {
  section <- parsed_report$data$read_quality_control
  if (!("paired_distance" %in% names(section))) {
    return(NULL)
  } else {
    figure <- section$paired_distance$figure_1
    return(c("title" = figure$figure_title,
             "x" = figure$x_axis_title,
             "y" = figure$y_axis_title))
  }
}

# Re-order the intervals for the paired distances by using the starting value of the interval.
order_paired_distances <- function(paired_distance) {
  distances <- unique(paired_distance$distance)
  starting <- as.numeric(sapply(strsplit(distances, split = " - "), function(l) l[1]))
  distances <- distances[sort.int(starting, index.return = TRUE)$ix]
  paired_distance$distance <- factor(paired_distance$distance, levels = distances)
  breaks <- distances[round(seq(from = 1, to = length(distances), length.out = 15))]
  return(list(data = paired_distance, breaks = breaks))
}

Using the above functions, the script below parses all the JSON reports found in the "exported
reports" folder, to build a read count statistics table (read_count_statistics), and a paired
distance histogram.

reports <- list.files("exported reports/", full.names = TRUE)
read_count_statistics <- data.frame()
paired_distance <- data.frame()
titles <- c(NA, NA, NA)
for (report in reports) {
  parsed_report <- fromJSON(report)
  read_count_statistics <- rbind(read_count_statistics,
    get_read_count_stats(parsed_report))
  paired_distance <- rbind(paired_distance,
    get_paired_distance(parsed_report))
  titles <- get_paired_distance_titles(parsed_report)
}

paired_distance <- order_paired_distances(paired_distance)

ggplot(paired_distance$data, aes(x = distance, y = number_of_reads, fill = sample)) +
  geom_bar(stat = "identity", position = "dodge") +
  scale_x_discrete(breaks = paired_distance$breaks, labels = paired_distance$breaks) +
  labs(title = titles["title"], x = titles["x"], y = titles["y"]) +
  theme(legend.position = "bottom")

You can try out the JSON export of RNA-Seq reports and the above script with the data included in
the tutorial Expression Analysis using RNA-Seq: http://resources.qiagenbioinformatics.
com/tutorials/RNASeq-droso.pdf

7.2.9 Graphics export

CLC Main Workbench supports two ways of exporting graphics:

- You can export the current view, either the visible area or the entire view, by clicking on the Graphics button (□) in the top Toolbar. This is the generally recommended route for exporting graphics for individual data elements, and is described in section 7.3.

- For some data types, graphics export tools are available from the main Export menu, which can be opened by clicking on the Export (□) button in the top Toolbar. These are useful if you wish to export different data using the same view in an automated fashion, for example by running the export tool in batch mode or in a workflow context. This functionality is described below.

Using export tools to export graphics

The following types of data can be exported using dedicated export tools:

- Alignments
- Heat maps
- Read mappings
- Sequences
- Tracks
- Track lists

The general actions taken are:
• Click on the **Export** button in the top Toolbar or choose the **Export** option under the File menu.

• Type “graphics” in the top field to see just a list of graphics exporters, and then select the one you wish to use. For example, if you wish to export an alignment as graphics, select "Alignment graphics" in the list.

• Select the data elements to be exported.

• Configure any relevant options. Detailed descriptions of these are provided below.

• Select where the data should be exported to.

Options available when exporting sequences, alignments and read mappings to graphics format files are shown in figure 7.9.

![Export Sequence graphics](image)

Figure 7.9: Options available when exporting sequences, alignments and read mappings to graphics format files.

The options available when exporting tracks and track lists to graphics format files are shown in figure 7.10.

The format and size of the exported graphics can be configured using:

• **Graphics format**: Several export formats are available, including bitmap formats (such as .png, .jpg) and vector graphics (.svg, .ps, .eps).

• **Width and height**: The desired width and height of the exported image. This can be specified in centimeters or inches.

• **Resolution**: The resolution, specified in the units of "dpi" (dots per inch).

The appearance of the exported graphics can be configured using:
CHAPTER 7. IMPORT/EXPORT OF DATA AND GRAPHICS

Figure 7.10: Options available when exporting tracks and track lists to graphics format files.

- **View settings**: The view settings available for the data type being exported. To determine how the data will look when a particular view is used, open a data element of the type you wish to export, click on the **Save View** button visible at the bottom of the Side Panel, and apply the view settings in the dialog that appears. View settings are described in section 4.6. Custom view settings will be available to choose from when exporting if the "Save for all <data type> views" option was checked when the view was saved.

- **Region restriction**: The region to be exported. For sequences, alignments and read mappings, the region is specified using start and end coordinates. For tracks and track lists, you provide an annotation track, where the region corresponding to the full span of the *first* annotation is exported. The rest of the annotations in the track have no effect.

### 7.2.10 Export history

Each data element in the **CLC Main 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 **CLC Main 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 7.11).

• 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 7.12).

• Select where the data should be exported to.

• Click **Finish**.

![Select export format](image1)

**Figure 7.11:** Select "History PDF" for exporting the history of an element as a PDF file.

![Export History PDF](image2)

**Figure 7.12:** When exporting the history in PDF, it is possible to adjust the page setup.

### 7.2.11 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 ![Folder Icon](image3), 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.
CHAPTER 7. IMPORT/EXPORT OF DATA AND GRAPHICS

Here, if you needed to recover the data later, you could put 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 (保存)

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 7.2.4.

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

File | Import (导入) | 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 (BLAST) | Manage BLAST Databases (管理)

7.3 Export graphics to files

CLC Main Workbench supports two ways of exporting graphics:

- You can export the current view, either the visible area or the entire view, by clicking on the Graphics button (视图) in the top Toolbar. This is the generally recommended route for exporting graphics for individual data elements, and is described below.
• For some data types, graphics export tools are available in the main Export menu, which can be opened by clicking on the Export button in the top Toolbar. These are useful if you wish to export different data using the same view in an automated fashion, for example by running the export tool in batch mode or in a workflow context. That functionality is described in section 7.2.9.

Exporting a view of data element to a graphics format file

To export a view of an open data element to a graphics file, click on the Graphics button ( ) in the top Toolbar, or choose Export Graphics from under the File menu.

How the data looks on the screen is how it will look in the exported file. Before exporting, the options in the Side Panel can be used to make adjustments as necessary.

For views that can be zoomed into or out of, you will be offered the choice of exporting the whole view or just the visible area (figure 7.13). For 3D structures, the section visible will always be exported, i.e. the equivalent to selecting to export just the visible area.

![Figure 7.13: The whole view or just the visible area can be selected for export.](image)

A view of a circular sequence, zoomed in so that you can only see a part of it, is shown in figure 7.14.

![Figure 7.14: A circular sequence, as it looks on the screen when zoomed in.](image)

When the option Export visible area is selected, the exported file will only contain the part of the
sequence that is visible in the view. The result of doing this for the view shown in figure 7.14 is shown in figure 7.15.

![Figure 7.15: The exported graphics file when Export visible area was selected.](image)

If you select Export whole view, the result would look like that shown in figure 7.16, where the part of the sequence not visible on screen is also exported.

![Figure 7.16: The exported graphics file when Export whole view was selected. The whole sequence is shown, not just the part visible on screen when the view was exported.](image)

You are prompted for the file format to use, and the name and location to save the output to.

For vector-based image formats, click on Finish to start the export. For bitmap-based image formats, click on Next to select a resolution level. You can click on Finish directly instead, in which case the image will be exported using the setting used the last time this functionality was run. If it has not been run before the default settings are used.

The available export formats are described in section 7.3.1.
7.3.1 File formats

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

<table>
<thead>
<tr>
<th>Format</th>
<th>Suffix</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portable Network Graphics</td>
<td>.png</td>
<td>bitmap</td>
</tr>
<tr>
<td>JPEG</td>
<td>.jpg</td>
<td>bitmap</td>
</tr>
<tr>
<td>Tagged Image File</td>
<td>.tif</td>
<td>bitmap</td>
</tr>
<tr>
<td>PostScript</td>
<td>.ps</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Encapsulated PostScript</td>
<td>.eps</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Portable Document Format</td>
<td>.pdf</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Scalable Vector Graphics</td>
<td>.svg</td>
<td>vector graphics</td>
</tr>
</tbody>
</table>

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.

To produce a high resolution image with all the details of a large element visible, e.g. a large phylogenetic tree or a read mapping, we recommend exporting to a vector based format.

If Screen resolution and High resolution settings show the same pixel dimensions, this can be because the maximum supported number of pixels has been exceeded.

**Parameters for bitmap formats** For bitmap files, clicking Next will display the dialog shown in figure 7.17.

![Figure 7.17: Parameters for bitmap formats: size of the graphics file.](image)

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 7.1.1 for more about importing external files into **CLC Main Workbench**.

**Parameters for vector formats** For PDF format, the dialog shown in figure 7.18 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 7.18: Page setup parameters for vector formats.](image)

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.

### 7.4 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 7.19, showing the conservation score of reads in a read mapping.

![Conservation graph](image)

**Figure 7.19:** 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 7.20 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.

![Export graph to CSV dialog](image)

**Figure 7.20:** 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"
```
7.5 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 "File system" or "<servername> (CLC Genomics Server)", as shown in figure 7.21. 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 7.21: Importing data using a Server.

If you choose the option "File system" 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 2.4.
7.6 Copy/paste view output

The contents of various element types can be copied and then pasted into programs outside of the Workbench. For example, tables from reports, table views of sequence lists, and tabular listings of folders, can be copied and then pasted into text editors or programs like Excel.

**Example:** Right click a folder in the Navigation Area and choose **Show | Content**. The folder contents are shown as a table in the viewing area. Select one or more of the rows and then copy them (Ctrl + C). That information can then be pasted into other programs (figure 7.22).

![Selected elements in a Folder Content view.](image)

Workflow designs can be copied as an image. To do this, select the elements in the workflow design (click in the workflow editor and then press keys Ctrl + A), then copy (Ctrl + C), and then paste where you wish the image to be placed, for example, in an email or presentation program. Workflows are described in detail in chapter 11.
Chapter 8

Data download

Contents

8.1 Search for Sequences at NCBI ................................................. 131
  8.1.1 NCBI search options .................................................. 132
  8.1.2 Handling of NCBI search results .................................. 133

8.2 Search for PDB Structures at NCBI ...................................... 134
  8.2.1 Structure search options .............................................. 134
  8.2.2 Handling of NCBI structure search results ....................... 135
  8.2.3 Save structure search parameters ................................ 137

8.3 Search for Sequences in UniProt (Swiss-Prot/TrEMBL) .............. 137
  8.3.1 UniProt search options ............................................... 137
  8.3.2 Handling of UniProt search results ............................... 139

8.4 Sequence web info .......................................................... 140

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

8.1 Search for Sequences at NCBI

Search for Sequences at NCBI supports searching the Entrez nucleotide, peptide and EST databases (https://www.ncbi.nlm.nih.gov/books/NBK44864/).

To start this tool, go to:

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

Select the nucleotide, protein or EST database using options at the top of the view (figure 8.1).

The search you just set up can be saved by doing one of the following:

- Choose Save As... from under the File menu, or
- Click on the tab of the search view and drag and drop it into a folder in the Navigation Area.
These actions save the search query. (It does not save the search results.)
This can be useful when you run the same searches periodically.

8.1.1 NCBI search options

Conducting a search using **Search for Sequences at NCBI** corresponds to using the same query terms on the NCBI website. For example, searching the nucleotide database corresponds to a search at https://www.ncbi.nlm.nih.gov/nucleotide/. A list of the entries matching the search terms is returned. Those of interest can then be selected and then downloaded, ready for downstream use in the Workbench.

Click on **Add search parameters** to add parameters to your search. The following are available:

- **All Fields** Searches for the terms provided in all fields of the NCBI database.
- **Organism**
- **Definition/Title**
- **Modified** Search for entries modified within the period specified from a drop-down list.
- **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**

Check the “Append wildcard (*) to search words” checkbox to indicate that the term entered should be interpreted as the first part of the term only. E.g. searching for “genom” with that box checked would find entries starting with that term, such as “genomic” and “genome”.

When you are satisfied with the parameters you have entered, click on the **Start search** button.
Additional search tips

Feature keys can be added when searching for nucleotide sequences. Writing `gene[Feature key] AND mouse` searches for one or more genes and where 'mouse' appears somewhere in the entry. For more information about how to use this syntax, see [http://www.ncbi.nlm.nih.gov/books/NBK3837/](http://www.ncbi.nlm.nih.gov/books/NBK3837/)

Where multiple terms are entered into a single field, the terms are usually treated as if they have an AND between them, which means that entries returned must contain all the specified terms. A notable exception is if multiple accessions are listed in an "All Fields" search field, in which case a results for each valid accession listed is returned.

You can add the words "AND", "OR" or "NOT" to the terms in a given field to further customize your search. Further details about searching Entrez sequence databases can be found at [https://www.ncbi.nlm.nih.gov/books/NBK44864/](https://www.ncbi.nlm.nih.gov/books/NBK44864/).

8.1.2 Handling of NCBI search results

A list of entries matching the search terms provided is returned. By default, the initial 50 results are returned, with 50 additional results loaded each time the More... button is clicked. This number can be configured in the Workbench Preferences (section 4).

Five pieces of information are presented for each entry:

- **Hit** The rank of the entry in the search result
- **Accession** The accession for that entry. Click on the link to open that entry’s page at the NCBI in a web browser.
- **Description** The definition line of the entry
- **Modification date** The date the entry was last updated in the database searched
- **Length** The length of the sequence

The columns to display can be configured in "Show column" tab of right hand, side panel settings. Select one or more rows of the table and use buttons at the bottom of the view to:

- **Download and Open** Sequences are opened in a new view after download is complete.
  
  You can also download and open sequences by dragging selected rows to a new tab area or by double-clicking on a row.

- **Download and Save** Sequences are downloaded and saved to a location you specify.
  
  You can also download and save sequences by selecting rows and copying them (e.g. using Ctrl + C), and then selecting a folder in the Navigation Area and pasting (e.g. using Ctrl + V).

- **Open at NCBI** The sequence entry page(s) at the NCBI are opened in a web browser.
The functions offered by these buttons are also available in the menu that appears if you right-click over selected rows. 

**Note:** The modification date on sequences downloaded can be more recent than those reported in the results table. This depends on the database versions made available for searching at the NCBI.

Downloading and saving sequences can take some time. This process runs in the background, so you can continue working on other tasks. The download process can be seen in the **Status bar** and it can be stopped, if desired, as described in section 2.4.

### 8.2 Search for PDB Structures at NCBI


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 8.2:

![Figure 8.2: The structure search view.](image)

**8.2.1 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](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.

### 8.2.2 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 8.3). 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.

![Image](image.png)

**Figure 8.3:** 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](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.

### 8.2.3 Save structure search parameters

The search you just set up can be saved by doing one of the following:

- Choose **Save As...** from under the File menu, or
- Click on the tab of the search view and drag and drop it into a folder in the Navigation Area.

These actions save the search query. (It does not save the search results.)

This can be useful when you run the same searches periodically.

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

**Search for Sequences in UniProt** searches the UniProt Knowledgebase (UniProtKB). UniProtKB contains UniProtKB/Swiss-Prot entries, which are manually curated, and UniProtKB/TrEMBL entries, which are annotated by automated systems.

To open the tool, go to:

**Download** | **Search for Sequences in UniProt**

One or more search terms can be entered, and a table of entries matching the search terms is returned. Rows in that table can be selected and the corresponding UniProtKB entries downloaded and opened, or downloaded and saved (figure 8.4).

Searching is described in more detail in section 8.3.1.

Working with results returned by a search is described in section 8.3.2.

#### 8.3.1 UniProt search options

**Database choice**
Select one of the 2 subsections of UniProtKB to search in, or select both to search all of UniProtKB.

- **Swiss-Prot** Searches among manually curated entries. These are the entries marked as "reviewed" in UniprotKB.
- **TrEMBL** Searches among computationally analyzed entries that have been annotated using automated systems. These are the entries marked "unreviewed" in UniprotKB.

**Search fields**

A single search field is presented by default. Click on "Add search parameters" to add more.

The following options are available:

- **All fields** Search for the term provided in all fields available at the UniProtKB website https://www.uniprot.org/.
- **Accession** Search for entry accessions.
- **Protein Names** Search for terms in protein names.
- **Gene Names** Search for terms in gene names.
- **Organism** Search for terms in organism names.
- **Created** Search for entries created within the period specified from a drop-down list.
- **Modified** Search for entries modified within the period specified from a drop-down list.
- **Protein existence.** Search for entries with the evidence level specified from a drop-down list.

When the **Append wildcard (\*) to search words** is checked, the search is broadened to include entries containing terms starting with text you provided.

Click on the **Start search** button to run the search.
Information about entries meeting all the conditions specified is returned in a table. No data is downloaded at this point. Working with these results, including downloading entries, is described in section 8.3.2.

**Saving a configured UniProt search**

The search you just set up can be saved by doing one of the following:

- Choose **Save As...** from under the File menu, or
- Click on the tab of the search view and drag and drop it into a folder in the Navigation Area.

These actions save the search query. (It does not save the search results.) This can be useful when you run the same searches periodically.

### 8.3.2 Handling of UniProt search results

#### The results table

By default, the first 50 entries found in UniProtKB that met all the search conditions are returned in the results table. Clicking on the **More...** button at the bottom right loads further results.

The default value of 50 can be changed. See chapter 4 for details.

For each entry, the following information is provided:

- **Hit** The position of the entry in the results. E.g. 1 for the first entry in the list returned, 2 for the second, and so on.
- **Accession** The accession of the entry. Clicking on the link opens the entry’s page at the UniprotKB website.
- **ID** The ID of the entry. Clicking on the link opens the entry’s page at the UniprotKB website.
- **Protein Names** The protein names and synonyms.
- **Gene Names** The names of the gene(s) encoding the protein.
- **Organism** The scientific name and synonyms of the source organism.
- **Created** The date the entry was created.
- **Modified** The date of the latest annotation update.
- **Length** The length of the canonical sequence.
- **Protein Existence** The level of evidence supporting the existence of the protein.
- **Pubmed Entries** The list of Pubmed IDs mapped to the entry. Clicking on the link opens a page listing these Pubmed entries.
- **Reviewed** Either "reviewed" for entries in Swiss-Prot, or "unreviewed" for entries in TrEMBL.
The columns displayed can be customized using in the side panel settings. See section 4.6 for details.

If you wish to open webpages for several entries at once, highlight the rows of interest and click on the Open at UniProt button.

**Downloading results from UniprotKB**

Entries from UniprotKB can be downloaded and saved, or downloaded and opened, directly in the viewing area. When opened directly, the entries are not saved until you take explicit action to do so (see section 2.1.2).

Downloading many large files may take some time. The download process can be stopped from under the Processes tab (see section 2.4).

To download and save entries, select the rows of interest and then do one of the following:

- Click on the Download and Save button. You will be prompted for a location to save the entries to.
- Right-click over a selected area and choose the option Download and Save from the menu presented.
- Drag the row(s) to a folder in the Navigation Area.
- Copy (Ctrl-C) to copy the entry information. Click on a folder in the Navigation Area and then paste (Ctrl-V).

The selected entries are downloaded from UniprotKB. Multiple entries selected at the same time are saved to a single protein sequence list.

To download and open entries directly in the viewing area, select the rows of interest and then do one of the following:

- Click on the Download and Open button.
- Right-click over a selected area and choose the option Download and Open from the menu presented.
- Drag the row(s) until the mouse cursor is next to an existing tab in the view area. When the mouse button is released, a new tab is opened, and the selected entries are downloaded and opened in that tab.

Double-clicking on a single row will download and open that entry.

### 8.4 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 12.5). 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 8.5):

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

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.
This section describes how to run tools, and how to handle and inspect results. We cover
launching tools for individual runs, as well as launching them in batch mode, where the tool is
run multiple times in a hands-off manner, using different input data for each run.

Launching workflows, individually or in batch mode, as well as running sections of workflows in
batch mode, are covered in chapter 11.

9.1 Running tools

Launching tools and workflows involves the same series of general actions:

- Data elements to be used in the analysis are selected.
- Any settings necessary for the tool/workflow to run are configured.
- The job is launched.
- Results are opened or saved when the job completes.

There are several ways to launch a tool or installed workflow:

- Double click on its name in the Toolbox tab in the bottom left side of the Workbench.
- Select it from the Toolbox menu at the top of the Workbench.
- Use the Quick Launch tool, described below.
Using the Quick Launch tool to start jobs

The Quick Launch tool, shown in figure 9.1, is started by clicking on the Launch button (Launch) in the toolbar. It can also be launched using the keyboard shortcut Ctrl + Shift + T (⌘ + Shift + T on Mac), or by going to the Toolbox menu at the top of the Workbench and selecting the top option, Launch (Launch).

![Quick Launch tool](image)

**Figure 9.1:** Tools and installed workflows can be quickly found and launched using the Quick Launch tool.

In the Quick Launch dialog, you can type terms in the text field at the top. This will filter for tools and installed workflows with matches to these terms in the name, description or Toolbox location. For tools where names have been changed between Workbench versions, searches using old names will still filter for the relevant tool. Using single or double quotes (’ or ”) will find a literal quote of the searched term.

In the example shown in figure 9.2, typing create shows a list of tools involving the word create. The arrow keys or mouse can be used for selecting and starting a tool from this list.

Click on the Favorites tab to see the subset of tools that are frequently used, or have been selected as favorites. See section 2.3.2 for further details.

Configuring and submitting jobs

When you open a tool or installed workflow, 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 9.1.1.

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 9.3
CHAPTER 9. RUNNING TOOLS, HANDLING RESULTS AND BATCHING

Figure 9.2: Typing in the search field at the top will filter the list of tools to launch.

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 9.3: 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 9.3.

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 9.4. Clicking on the Reset button resets all parameters in that step to their default values.

9.1.1 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 9.5.

- **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.

Figure 9.4: An example of a "Set parameters" window.

Figure 9.5: When logged into the CLC Server, you can select where a job should be run.
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.

The remaining wizard steps are the same whether you are launching a job on a **CLC Workbench** or a **CLC Server**, with two minor differences: when running on **CLC Server**, results are always saved, and a log of the job is always created and saved alongside the results.

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 workflows, this includes creating and saving a workflow result metadata table. By contrast, for a single analysis run on the Workbench, you can normally choose whether to **Open** or **Save** results, and you can choose whether to create a workflow result metadata table.

- 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 2.4.

### 9.2 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 9.6). You can save to an existing area or create a new folder to save the results into.

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 **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:
Figure 9.6: Specify where to save the results of an analysis.

- 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 9.7.

Figure 9.7: 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.

9.3 Batch processing

Batch processing refers to running an analysis multiple times, once per batch unit. For example, if you have 10 sequence lists and wish to run 10 mapping analyses, one per sequence list, you could launch all 10 analyses by setting up one batch job. Here, each sequence list would be a "batch unit".
This section focuses on batch processing when using **individual tools**. Further details about batch processing of workflows is provided in section 11.3 and section 11.3.3.

**Batch mode**

Batch mode is activated by clicking the **Batch** checkbox in the dialog where the input data is selected (figure 9.8).

![Batch mode](image)

Figure 9.8: When launching an analysis in **Batch mode**, individual elements and/or folders can be selected. Here, a single folder that contains both elements and subfolders of elements has been selected.

In Batch mode, the analysis is run once per batch unit. A batch unit consists of the data elements to be analyzed together. A batch unit can be a single data element, or can consist of multiple data elements.

**Batch units are made up of:**

- Each data element selected in the launch wizard.
- Elements and folders within a folder selected in the launch wizard, where:
  - Each data element contained directly within that selected folder is a batch unit.
  - Each subfolder directly under the selected folder is a batch unit. I.e. all elements within that subfolder are analyzed together.
  - Elements within more deeply nested subfolders (e.g. subfolders of subfolders of the originally selected folder) are not used in the analysis.
- Elements with associations to a CLC Metadata Table selected in the launch wizard. Each row in the CLC Metadata Table is a batch unit. Data elements associated with a row, of a type compatible as input to the analysis, are the default contents of a batch unit. See figure 9.9 and figure 9.10.

**Batch overview**
When the Batch box is checked, a CLC Metadata Table can be selected as input.

Data associated with each row in a CLC Metadata Table, of a type compatible with that analysis, make up the default content of batch units.

In the batch overview step, the elements in each batch unit can be reviewed, and refined based on their names using the fields Only use elements containing and Exclude elements containing.

In figure 9.11, the batch units, i.e. those elements and folders directly under the folder selected in figure 9.8, are shown. In each batch unit, data elements that could be used in the analysis are listed on the right hand side. Some batch units contain more than one data element. Those data elements would be analyzed together. To limit the analysis to just sequence lists containing trimmed sequences, the term “trim” has been entered into a filter field near the bottom.

Folders that do not contain any elements compatible with the analysis are not shown in the batch overview.

Organization of the results

The options for where to save analysis outputs are shown in figure 9.12.

The available options are:

- **Save in input folder** Save all outputs into the same folder as the input data. For batch units defined by folders, the results of each analysis are saved into the folder with the input data. If the batch units were individual data elements, results are put into the same folder as the input elements.

- **Save in specified location** You will be prompted in the next step to select a folder where the outputs should be saved to. The Create subfolders per batch unit checkbox allows you to specify whether subfolders should be created to store the results from each batch unit:
Figure 9.11: Overview of the batch units (left) and the input elements defined by each batch unit (right). By default, all elements that can be used as inputs are listed on the right (top). By entering terms in the filter fields, the list of elements in the batch units can be refined. Here, only sequence lists including trimmed sequences will be included (bottom).

Figure 9.12: Options for saving results when an analysis is run in Batch mode.

- When checked results for each batch unit are written to a newly created subfolder under the folder you select in the next step. A subfolder is created for each batch unit. (This is the default option.)
- When unchecked, results from all batch units are written to the folder you select in the next step.

The log file

In the final wizard step there is an option to Create a log. When checked, a log containing
information about all the batch units will be created. This log includes the term "combined log" in its name. A log is also created for each individual batch unit.

**Batch unit processes**

When the job is running, there is one "master" process representing the overall batch job, and a separate process for each batch unit.

On a **CLC Workbench**, the batch units are executed sequentially - one batch unit at a time. This avoids overloading the computer.

On a **CLC Server**, all the processes are placed in the queue, and the queue takes care of distributing the jobs. If there are multiple job nodes or grid nodes, batch units may be processed in parallel.

**Stopping a batch run**

To stop the whole batch run, stop the "master" process.

On a **CLC Workbench**, find the master process in the Processes tab in the bottom left side. Click on the little triangle on the right hand side of the master process and choose the option **Stop**.
Chapter 10

Metadata

Contents

10.1 Creating metadata tables ................................................. 155
  10.1.1 Importing metadata .............................................. 155
  10.1.2 Creating a metadata table directly in the Workbench .......... 158
10.2 Associating data elements with metadata ............................. 161
  10.2.1 Associate Data Automatically .................................. 162
  10.2.2 Associate Data with Row ...................................... 164
10.3 Working with data and metadata ...................................... 165
  10.3.1 Finding data elements based on metadata ....................... 165
  10.3.2 Viewing metadata associations ................................... 166
  10.3.3 Removing metadata associations ................................. 167
  10.3.4 Identifying metadata rows without associated data ............. 168
  10.3.5 Editing Metadata tables ....................................... 168
10.4 Moving, copying and exporting metadata .............................. 172

Metadata refers to information about data. In the context of the CLC Main Workbench, this usually means 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.

Examples in this chapter refer to tools present in the CLC Genomics Workbench, but the principles apply to other CLC Workbenches.

What is metadata used for? Core uses of metadata in CLC software include:

- Defining batch units when launching workflows in batch mode, described in section 11.3.2.
- Distributing data to the relevant input channels in a workflow when using Collect and Distribute elements, described in section 11.2.4.
- Finding and selecting data elements based on sample information (in a CLC Metadata Table). Workflow Result Metadata tables are of particular use when reviewing results generated by workflows run in batch mode and are described in section 11.3.1.

**Metadata tables**

An example of a CLC Metadata Table in the *CLC Main Workbench* is shown in figure 10.1. Each column represents a property of a sample (e.g., identifier, height, age, treatment) and each row contains information relevant to a sample. A single column can be designated the key column. That column must contain unique entries.

![Figure 10.1: A simple metadata table, with the key column highlighted in blue.](image)

Each row can have associations with one or more data elements, such as sequence lists, expression tracks, variant tracks, etc. Associating data elements with relevant metadata rows, automatically or manually, is covered in section 10.2.

Information from an Excel, CSV or TSV format file can be imported into a CLC Metadata Table, as described in section 10.1.1. CLC Metadata Tables are also generated by workflows, as described in section 11.3.1.

**Metadata Elements table**

Data elements with an association to a row in a CLC Metadata Table can be listed by selecting the rows of interest and clicking on the **Find Associated Data** button. The Metadata Elements table opens, with a table of information about elements with associations to the selected rows (figure 10.2).

When a data element is associated with a metadata row, the outputs of analyses involving that data usually inherit the metadata association automatically. For example, if a sequence list with an association to a CLC Metadata Table row is used as input to analyses, results of these analyses may also be associated with that row (figure 10.2).

Inheritance of associations to metadata requires that a single association can be unambiguously identified for an output when a tool is run. If an output is derived from two or more inputs with different metadata associations, then no association will be inherited.
10.1 Creating metadata tables

CLC Metadata tables can be created in several ways:

- Import metadata from an Excel, CSV or TSV format file using the Import Metadata tool, described in section 10.1.1.
- As an output from workflows (Workflow Result Metadata), described in section 11.3.1.
- By selecting rows of an existing CLC Metadata Table, right-clicking over the selected rows and choosing the option Create New Metadata Table... ( ). Data associated with those rows will also be associated to the rows of the new CLC Metadata Table.

10.1.1 Importing metadata

The Import Metadata tool

To import metadata into a CLC Metadata Table, go to:

File | Import ( ) | Import Metadata ( )

The first column in the selected file must have unique entries. That column will be designated as the key column. A different column can be specified as the key column later. For the optional step for association of data elements to work, the first column must contain entries that can be matched with the relevant data element names.

Select the Excel, CSV or TSV format file with metadata to be imported. The rows in that file are displayed in the Metadata preview window (figure 10.4).

The format of the columns should reflect the column contents. The designated format can be changed from within the Metadata Table editor, as described in section 10.3.5. There, you...
CHAPTER 10. METADATA

Figure 10.3: Select rows in a CLC Metadata table, right-click and choose "Create New Metadata Table..." to create an element with just those rows.

can change the column data types (e.g. to types of numbers, dates, true/false) and you can designate a new key column.

Figure 10.4: Rows being imported from a file containing metadata are shown in the Metadata preview table.

**Associating metadata with data (optional)**  The "Associate with data" wizard step (figure 10.5), is optional. To proceed without associating data to metadata, click on the **Next** button. Associating data with metadata can be done later, as described in section 10.2.

To associate data with the metadata:

- Click on the file browser button to the right of the **Location of data** field
• Select the data elements to be associated.
• Select the matching scheme to use: Exact, Prefix or Suffix. These options are described in section 10.2.1.

![Image of Import Metadata window]

Figure 10.5: Three data elements are selected for association. The "Prefix" partial matching scheme is selected for matching data element names with the appropriate metadata row, based on the information in the Sample ID column in this case.

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.

You can then select where you wish the metadata table to be saved and click on Finish.

The associated information can be viewed for a given data element in the Show Element Info view (figure 10.6).

![Image of Show Element Info view]

Figure 10.6: Metadata associations can be seen, edited, refreshed or deleted via the Show Element Info view.
10.1.2 Creating a metadata table directly in the Workbench

Creating CLC Metadata tables from scratch within the CLC Main Workbench is described in this section.

See also section 10.1.1 for importing information from Excel to create a CLC Metadata Table. If your analysis is contained in a workflow, a CLC Metadata Table can be created automatically. This is described in section 11.3.1.

To create a CLC Metadata Table manually, 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 Setup Table at the bottom of the view (figure 10.7).

![Figure 10.7: Dialog used to add columns to an empty Metadata Table.](image)

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.

To import the table from a file, click on Setup Structure from File. In the dialog that appears (figure 10.8), 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 (;).

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.
Populating the table  Click on Manage Data button at the bottom of the view (figure 10.9).

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 10.10.

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 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 use the Java patterns for date and time formatting. Meaning of some of the symbols:

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

Examples of using this:
### 10.2 Associating data elements with metadata

Each row in a metadata table can be associated with one or more data elements, such as sequence lists, expression tracks, variant tracks, etc. Each data element can be associated with one row of a given metadata table. Once data elements are associated with rows of a metadata table, it is then possible to use that metadata table to find data elements that share particular attributes, launch analyses like expression analyses where sample attributes are key, define batch units such that an analysis runs once per sample, or to group samples together according to their attributes when running certain types of workflows.

Each association has a "Role" label assigned to the associated element, which can be used to indicate the nature of the data element. For example, a newly imported sequence list could be given a role like "Sample data", or "NGS reads".

Associating data with metadata rows can happen in several ways, depending on the circumstances:

- By default, when input data for an analysis is associated with metadata, the results will inherit any unambiguous association. Appropriate role labels are assigned by the analysis tool. For example, a read mapping tool will assign the role "Unmapped reads" to a sequence list of unmapped reads that it produces.

- By default outputs from a workflow are associated with the relevant metadata rows in workflow results metadata tables. In these tables, the role assigned is always "Result data".

- Manually triggering data associations, either through matching the metadata key column entries with data element names, or by specifying the data element to associate with a given row. Here, roles to apply are chosen by you when triggering the associations.

The rest of this section describes this last point, where you associate data elements to metadata. To do this, open a metadata table, and then click on the **Associate Data** button at the bottom of the Metadata Table view. Two options are available:
• **Association Data Automatically** Associations are set up based on matches between metadata key column entries and data elements names in a specified location of the Navigation Area. This option is only available if a key column has been specified. See section 10.2.1.

• **Associate Data with Row** Manually make associations row by row, by selecting a row of the metadata and a particular data element in the Navigation Area. Here, information in the metadata table does not need to match data element names. This option is also available when right-clicking a row in the table. section 10.2.2.

### 10.2.1 Associate Data Automatically

When using the **Associate Data Automatically** option, associations are created based on matching the name of the selected data elements with the information in the key column of a metadata table previously saved in the Navigation Area. Matching is done according to three possible schemes: Exact, Prefix and Suffix (see section 10.2.1).

To associate data automatically, click the **Associate Data** button at the bottom of the Metadata Table view, and select **Associate Data Automatically**.

Select the data the tool should consider when setting up metadata associations (figure 10.11).

![Figure 10.11: Select data elements to be associated to a CLC Metadata Table.](image)

In the Association setup step, you specify whether the matching of the data element names to the entries in the key column should be based on exact or partial matching (described below). A preview showing how elements are matched to metadata rows using the selected matching scheme is shown in the wizard (figure 10.12).

You also specify a role for each element. The default role provided is "Sample data". You can specify any term you wish.

After the job has run, data associations and roles are saved for all the selected data elements where the name matches a key column entry according to the selected matching scheme.

**Note:** Data elements selected that already have associations with the CLC Metadata Table will have their associations updated to reflect the current information in the CLC 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. 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.

- **Prefix** - data elements with names partially matching a key will be associated: here the first whole part(s) of a name must match a key entry in the metadata table for an association to be established. This option is explained in more detail below.

- **Suffix** - data elements with names partially matching a key will be associated: here the last whole part(s) of a name 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 (with the Prefix option) or the last (with the Suffix option) 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 they key entry had been Sample23, no match would be identified, because they 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 10.13 (i.e., ETC-001, ETC-002, . . . , ETC-013),

<table>
<thead>
<tr>
<th>Data Element</th>
<th>Key</th>
<th>Reason for association</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETC-001 (Reads)</td>
<td>ETC-001</td>
<td>Key ETC-001 matches the first part of the name</td>
</tr>
<tr>
<td>ETC-001 un-m...</td>
<td>ETC-001</td>
<td></td>
</tr>
<tr>
<td>Single</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ETC-001 un-m...</td>
<td>ETC-001</td>
<td></td>
</tr>
<tr>
<td>Paired</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ETC-002</td>
<td>ETC-002</td>
<td>Key ETC-002 matches the whole name</td>
</tr>
<tr>
<td>ETC-003</td>
<td>None</td>
<td>No keys match this data element name</td>
</tr>
<tr>
<td>ETC-005</td>
<td>ETC-005</td>
<td>Key ETC-005 matches the whole name</td>
</tr>
<tr>
<td>ETC-005-1</td>
<td>ETC-005</td>
<td>Key ETC-005 matches the first part of the name</td>
</tr>
<tr>
<td>ETC-006-5</td>
<td>ETC-006</td>
<td>Key ETC-006 matches the first part of the name</td>
</tr>
<tr>
<td>ETC-007</td>
<td>None</td>
<td>No keys match this data element name</td>
</tr>
<tr>
<td>ETC-007 (mapped)</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>ETC-008</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>ETC-008 (report)</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>ETC-009</td>
<td>ETC-009</td>
<td>Key ETC-009 matches the whole name</td>
</tr>
</tbody>
</table>

10.2.2 Associate Data with Row

The Associate Data with Row option is best suited for association of a few metadata tables to a few data elements. This type of association does not require a key column in the metadata table, nor a particular 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. Then either click the Associate Data button at the bottom of the Metadata Table view, or right-click on the selected metadata row and choose the Associate Data with Row option (as seen in figure 10.13).

Figure 10.13: 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.

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.

**10.3 Working with data and metadata**

**10.3.1 Finding data elements based on metadata**

Data elements associated with rows of the metadata table can also be found from within a Metadata Table view. From there, it is possible to highlight elements in the Navigation Area and launch analyses on selected data.

Relevant metadata tables can be found using the Quick Search box, described in section 3.4.1.

To find data elements associated with selected metadata rows in a metadata table:

- Select one or more rows of interest in the metadata table.
- Click on the Find Associated Data button 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 10.14).

![Metadata Table with search results](image)

Figure 10.14: 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 10.14, 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 unmapped 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 3.4. 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](http://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Rebuilding_index.html).

### 10.3.2 Viewing metadata associations

Metadata associations for a data element are shown using the Element info view, as in figure **10.15**.

To show Element Info,

![Element Info view with a metadata association](Figure 10.15: 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 12.4.

### 10.3.3 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 **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 10.16). Alternatively, use the Delete key on the keyboard, or on a Mac, the fn and backspace keys at the same time.

Figure 10.16: *Removing metadata associations to two data elements via the Metadata Elements table.*

Metadata associations can also be removed from within the Element info view for individual data elements, as described in section 10.3.2.
When an metadata association is removed from a data element, this update to the data element is automatically saved.

10.3.4 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 10.2.

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.

10.3.5 Editing Metadata tables

CLC Metadata Tables can be edited using functionality under the Edit Table... button at the bottom of the Metadata Table editor (figure 10.17) and by right-clicking on a row or rows and selecting from the options presented (figure 10.18). Information can be added directly, or can be take from another CLC Metadata Table or an Excel, CSV or TSV format file.

Edit existing rows in a CLC Metadata Table

A dialog to edit entries in a CLC Metadata Table can be opened by:

- Clicking on the Edit Table... button and selecting Edit Entries.. ( Edit Entry.. ( )).
- Double clicking on an entry.
- Selecting an entry and hitting the Return key.

Navigate between entries using the buttons on the right. Modifications made take effect as you navigate to another row, or if you close the dialog using Done.

Right-click on an individual row in the table and select the Edit Entry.. ( Edit Entry.. ( ) option to edit just that entry. An option to delete rows is also in this menu: Delete Row(s) ( Delete Row(s) ( )).

Import further rows into a CLC Metadata Table
Figure 10.17: Click on the Edit Table... button to open a menu with options for adding, editing or removing information in a CLC Metadata table.

Figure 10.18: Right-click on selected rows of a CLC Metadata Table to open a menu actions that can be taken.

More information can be imported into an existing CLC Metadata table by clicking on the Edit Table... button and selecting Import Metadata... ( ).

Information can be supplied in an Excel, CSV or TSV file, by clicking on the standard folder icon ( ), or from another CLC Metadata Table, by clicking on the browse folder ( ) icon.

For CLC Metadata Tables with a key column, you choose whether new information should be added to existing entries, and whether rows should be added for new entries (figure 10.21).
Matching is based on column names.

If there is no key column designated in the original CLC Metadata Table, as is the case in Workflow Result Metadata tables, then data from the new source can only be entered into new rows. A key column can be specified using the Configure Columns functionality described in the Add or edit columns in a CLC Metadata Table section below.

Figure 10.19: Additional information can be imported to an existing CLC Metadata table. You can choose whether new information should be added to existing entries, and whether rows should be added for new entries. The columns to import can also be specified.

Add and delete rows manually

Rows can be added to the end of the CLC Metadata table by clicking on the Edit Table... button and selecting Add Entries... ( ). This option is also available in the menu that opens when you right-click on a row in the table (figure 10.18).

The option Duplicate Entry, available when you right-click on a row in the table, will duplicate that row and put it at the bottom of the table. When a key column has been designated, you will be prompted for a new value for that column for the new row. Data with an association to the original row will not automatically have an association to the new row. (A given data element can only have one association to a given CLC Metadata Table.)

Individual rows can also be added using the ( ) button, which inserts a new row after the current one.

Rows may be deleted using the ( ) button.

The ( ) and ( ) buttons are used to undo and redo changes respectively.

Add or edit columns in a CLC Metadata Table

Existing columns can be edited, and new columns added, by clicking on the Edit Table... button and selecting Configure columns....

A dialog opens with information about the left-most column in the CLC Metadata Table (figure 10.21). Use the buttons on the right to navigate to other columns and to add or delete
columns.

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.

Individual columns can be added using the ( and ) buttons, which insert new columns before and after the current column respectively. Columns may be deleted using the ( button.

The ( ) and ( ) buttons are used to undo and redo changes respectively.

Figure 10.20: When adding a new column, a name, description and data type is specified. If it should become the key column, the Key column box should be checked. Use the buttons on the right to navigate to other columns or add further new columns.

Figure 10.21: The Name column has been designated as the key column.

For each column, the following can be configured:

- **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.** Any column containing only unique values can be designated as the key column. If a table already has a key column, this option is disabled for other columns. Information in the key column is used when automatically creating associations from data elements, described in (section section 10.2.1).

- **Type.** The type of value allowed. The default data type for columns on import is text, but this can be edited to the following types:
– **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.

### 10.4 Moving, copying and exporting metadata

**Moving and copying metadata**

This section focuses on what happens to data associations when copying metadata tables. This information also pertains to moving metadata tables *between* File Locations, as this is equivalent to a copy action.

To copy a metadata table and the data associated with it, such that the new data element copies are associated with the copy of the metadata table, *select both the metadata table and the data elements and copy them in a single operation*. Of note here:

- The data element copies will have associations with the new copy of the metadata table. The original elements keep their associations with the original metadata table.
- If a metadata table is copied but data elements with associations to it are not also copied in that action, those data elements will be associated with both the copy and the original metadata table.
- If data elements with associations to metadata are copied, but no metadata table is involved in the same copy action, each data element copy will be associated to the same metadata as the original element.

If a metadata table and some, but not all, data elements with associations to it, are copied in a single action, then:

- The data element copies will have associations to the copy of the metadata table, while the original elements (that were copied) remain associated with the original metadata table.
- Elements with associations to the original metadata table that were not copied will have associations to both the original metadata table and the copy. However, if these data elements are later copied (in a separate copy operation), those copies will only be associated with the original metadata table. If they should be associated with the copy of the metadata table, those association must be added as described in section 10.2.
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 7.2 for more information.
Chapter 11

Workflows

Contents

11.1 Creating and editing workflows ........................................... 175
  11.1.1 Adding elements to a workflow .................................... 176
  11.1.2 Connecting workflow elements .................................... 177
  11.1.3 Ordering inputs .................................................... 179
  11.1.4 Validating a workflow ............................................. 180
  11.1.5 Viewing the flow of elements in a workflow ................. 182
  11.1.6 Adjusting the workflow layout ................................... 182
  11.1.7 The Configuration Editor view ................................... 183
  11.1.8 Snippets in workflows ............................................ 183
  11.1.9 Customizing the Workflow Editor ............................... 188

11.2 Workflow elements ....................................................... 193
  11.2.1 Anatomy of workflow elements .................................... 193
  11.2.2 Basic configuration of workflow elements .................... 195
  11.2.3 Configuring input and output elements ......................... 199
  11.2.4 Control flow elements ............................................. 203
  11.2.5 Track lists as workflow outputs ................................. 208
  11.2.6 Input modifying tools ............................................ 208

11.3 Launching workflows individually and in batches ................... 209
  11.3.1 Workflow Result Metadata tables ............................... 210
  11.3.2 Running workflows in batch mode ............................... 212
  11.3.3 Running part of a workflow multiple times .................. 214

11.4 Advanced workflow batching ........................................... 217
  11.4.1 Batching workflows with more than one input changing per run 218
  11.4.2 Multiple levels of batching ..................................... 220

11.5 Managing workflows ..................................................... 220
  11.5.1 Updating workflows ............................................ 223
  11.5.2 Creating a workflow installation file ......................... 226
  11.5.3 Installing a workflow ........................................... 230
Workflows contains tools linked together, making up an analysis pipeline.

Data from CLC locations can be selected as input, as can raw data in supported standard formats. Results can be saved to CLC locations for viewing and downstream analysis, or exported to standard formats and saved to a location of your choice.

The flow of data through a workflow can be defined in various ways, making it simple to execute complex analyses consistently. Launching workflows in Batch mode, the same analysis is run on many samples using a single launch action.

Workflows are created and customized using a graphical workflow editor (figure 11.1). Workflows can be saved and re-opened, just like data elements. You can run workflows directly from the workflow design, or install them and run them from the Toolbox (see section 11.5.2).

Figure 11.1: A workflow consists of connected tools, where the output of one tool is used as input for another tool. Here a workflow is open in the Workflow Editor.

Examples in this chapter use tools from the CLC Genomics Workbench. Some of these are not available in the CLC Main Workbench. However, the principles described apply equally to tools in CLC Main Workbench.

11.1 Creating and editing workflows

Workflows are created and edited using the Workflow Editor.
To create a new workflow, click on the **Workflows** button (wę) in the Toolbar and then select "New Workflow" (wę).

Alternatively, use the menu option:

File | New | Workflow (wę)

To open a copy of an installed workflow in the Workflow Editor, right-click on the workflow in the Toolbox and choose the option **Open Copy of Workflow**. Customizing existing workflows rather than building new workflows from scratch, can save much time.

To copy the image of a workflow design, select the elements in the workflow design (click in the workflow editor and then press keys Ctrl + A), then copy (Ctrl + C), and then paste where you wish the image to be placed, for example, in an email or presentation program.

### 11.1.1 Adding elements to a workflow

Elements are the building blocks of workflows. They are used to define the inputs, the outputs, all the analysis steps to be done, and can be used to control the way data flows through the workflow. Workflow elements are defined in detail in section 11.2. Here we focus on adding them and connecting them to each other within a workflow.

To add elements to a workflow:

- Drag tools from the **Toolbox** in the bottom, left panel of the Workbench into the canvas area of the Workflow Editor, or
- Use the **Add Element** dialog (figure 11.2). The following methods can be used to open this dialog:
  - Click on the **Add Element** (wę) button at the bottom of the Workflow Editor.
  - Right-click on an empty area of the canvas and select the **Add Element** (wę) option.
  - Use the keyboard shortcut Shift + Alt + E.

![Add Elements](image)

Figure 11.2: **Adding elements to a workflow.**
Select one or more elements and click on **OK**. Multiple elements can be selected by keeping the Ctrl key (⌘ on Mac) depressed while selecting them.

- Use one of the relevant options offered when right-clicking on an input or output channel of a workflow element, as shown in figure 11.3 and figure 11.4.

![Connection options](image)

**Figure 11.3:** Connection options are shown in menus when you right click on an input or output channel of a workflow element.

![Right-click](image)

**Figure 11.4:** Right clicking on an output channel brings up a menu with relevant connection options.

- **Connect to Workflow Input** or **Connect to Configured Workflow Input** and
- **Use as Workflow Output** to add data to be processed or saved.
- **Add Element to be Connected...** to open the Add Elements pop-up dialog described above.
- **Connect to Iterate** and **Connect to Collect and Distribute** to create an iterative process within a workflow.

Once added, workflow elements can be moved around on the canvas using the 4 arrows icon (➡️) that appears when hovering on an element.

Workflow elements can be removed by selecting them and pressed the delete key, or by right-clicking on the element name and choosing **Remove** from the context specific menu, as shown in figure 11.5.

### 11.1.2 Connecting workflow elements

Setting up connections between elements is the key to controlling where data flows. Elements are connected via their output channels and input channels. These channels are type dependent:
only those with a matching type can be connected. Hovering the mouse cursor over an input or output channel will show a tooltip with information like the data type required.

Compatible input and output channels can be connected in various ways:

- Click on an output channel, keep the mouse button depressed, and move the cursor to the desired input channel. A green border around the input channel name indicates when the connection has been made and the mouse button can be released. An arrow is drawn, linking the channels, as shown in figure 11.6.

- Use the Connect <channel name> to... option in the right-click menu of an input or output channel. Hover the cursor over this option to see a list of elements in the workflow with compatible channels. Hovering the cursor over any of these items then shows the particular channels that can be connected to, as shown in figure 11.7.

A given output channel can supply its data to multiple other workflow elements, and workflow elements can accept data as input from more than one output channel. This is illustrated in figure 11.8.

Input and output channels are discussed further in section 11.2.
11.1.3 Ordering inputs

Two different ways to order workflow inputs are available. In both cases, an Order Inputs dialog like that shown in figure 11.9 is used. An item is moved up and down in the list by selecting it and clicking on the up or down arrow on the right.

Workflow inputs can be ordered by using:

- **Order Workflow Inputs**.... Right click on any blank area on the Workflow Editor canvas to bring up a menu with this option. This sets the order of the wizard steps prompting for the relevant input data when the workflow is launched. This ordering is reflected by a number in front of the Workflow Input element name in the workflow design. That number can also be used when configuring output element naming patterns, as described in section 11.2.3.

- **Order Inputs**.... Right click on an input channel with more than one connection to it to bring
up a menu with this option enabled. This is the order that inputs to this input channel should be processed. This ordering is reflected by a number on the arrow connecting to the input channel. This is particularly useful when considering data visualization. For example, when connecting inputs to a Track List element, this is the order the constituent tracks will be in. This is described further in section 11.2.5.

See figure 11.10 for an illustration of the effects of these 2 ordering methods.

Figure 11.9: The Order Inputs dialog is used to specify the ordering of workflow inputs.

Figure 11.10: Two levels of input ordering are available. Using Order Workflow Inputs..., the order that inputs are prompted for in the wizard was set to B, A, C. Using Order Inputs..., the order the inputs were processed was set to C, A, B. Here, this means the tracks will appear in the Track List in the order C, A, B.

11.1.4 Validating a workflow

At the bottom of the view, there is a text with a status of the workflow (see figure 11.11). It informs you about about the actions needed to finalize the workflow.

Figure 11.11: A workflow is constantly validated at the bottom of the view.

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 are done to verify that the workflow is valid, and specific warning messages will inform of eventual inconsistencies. The validation may contain several lines of text, and it is possible to scroll to see all lines. If one of the errors pertain to a specific element in the workflow, clicking the error will highlight this element.

Once these have been addressed, 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 11.2.2), there will be a dialog asking for this in the workflow launching wizard.

The workflow shown in figure 11.12 has not yet been saved, which is noted by the asterisk in the tab name and the message in red text below the workflow. If the workflow had been successfully saved, the text under the canvas would be in green text saying "Validation successful".

Figure 11.12: This copy of an installed workflow is available for editing in the Workflow Editor. It has not yet been saved for later use, as indicated by the asterisk in the tab name and the warning text below the workflow.
11.1.5 Viewing the flow of elements in a workflow

Following the path through a workflow from a particular element can help when authoring complex workflows. To highlight the path through the workflow from a particular element, right-click on the element name and select the Highlight Subsequent Path option from the context specific menu (figure 11.14). Select Remove Highlighting Subsequent Path to remove the highlighting.

11.1.6 Adjusting the workflow layout

The layout of elements within a workflow can be adjusted manually or automatically.

- **Manually**: Select one or more workflow elements and then, with the left mouse button depressed, drag these elements to where you want them to be on the canvas.

- **Automatically**: Right-click anywhere on the canvas and choose the option "Layout" (figure 11.15), or use the quick command Shift + Alt + L. The layout of all connected elements in the workflow will be adjusted.
See also section 11.1.9 for information about the **Auto Layout** setting. When enabled that setting causes the layout to be adjusted automatically every time an element is added and connected.

![Workflow diagram](image)

**Figure 11.15**: The alignment of workflow elements can be improved using the "Layout" function.

### 11.1.7 The Configuration Editor view

The **Configuration Editor** view allows the configuration of all of the settings in a workflow through a single window (figure 11.16). This can be more convenient than configuring each workflow element individually, especially for experienced workflow authors.

Click on the (antu) icon located in the lower left corner of the Workflow Editor to open this view.

### 11.1.8 Snippets in workflows

When creating a new workflow, you will often have a number of connected elements that are shared between workflows. These components are called snippets. Instead of building workflows from scratch it is possible to reuse components of an existing workflow.

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 11.17.

When you have clicked on "Install as snippet" the dialog shown in figure 11.18 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
Figure 11.16: Use the Configuration Editor to edit configurable parameters for all the tools in a given Workflow.

the selected elements and save it in the snippet or to only save the elements in their default configuration.

Click **OK**. This will install your snippet and the installed snippet will now appear in the **Side Panel** under the "Snippets" tab (see figure 11.19)

Right-clicking on the installed snippet in the **Side Panel** will bring up the following options (figure 11.20):

- **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.
- **Uninstall**. Removes the snippet.
- **Export**. Exports the snippet to ones computer, allowing to share it.
- **Migrate**. Updates the snippet (if required).

If you right-click on the top-level folder you get the options shown in figure 11.21:

- **Create new group**. Creates a new folder under the selected folder.
- **Remove group**. Removes the selected group (not available for the top-level folder)
Figure 11.17: The selected elements are highlighted with a red box in this figure. Select "Install as snippet".

- **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.

**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 element" option that is shown in figure 11.22.
Figure 11.18: 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.

Figure 11.19: When a snippet is installed, it appears in the Side Panel under the "Snippets" tab.
CHAPTER 11. WORKFLOWS

Figure 11.20: Right-clicking on an installed snippet brings up a range of different options.

Figure 11.21: Right-clicking on the snippet top-level folder makes it possible to manipulate the groups.

Figure 11.22: Snippets can be added to a workflow in the workflow editor using the ‘Add Element’ button found in the lower left corner.
11.1.9 Customizing the Workflow Editor

The Workflow Editor side panel (figure 11.23) contains settings that can make navigating through and editing workflows easier. They can be particularly useful when working with large workflows.

The Workflow Editor side panel contains the following:

**Minimap** A zoomed-out overview of the workflow. The darker grey box in the minimap highlights the area of the workflow visible in the editor. Drag that box within the minimap to quickly navigate to a specific area in the editor. The location of this dark grey box is updated when you navigate to another area of the workflow.
**Find** Search for workflow elements based on names.

After you enter a term and click on the Find button, the number of elements found with the term in their name is reported below the search field. The elements identified are highlighted in the workflow editor (figure 11.24). When multiple elements were found, the view will update so the first element found is visible in the editor. Click on the Find button again to bring the second element into view. Click on Find again to bring the third element into view, and so on.

![Workflow Editor with Find Tool](image)

**Figure 11.24:** Two elements with names including the term "venn" were found using the Find tool in the side panel. Both are visible in this view, with the first element found highlighted.

**Grid** Customize the spacing, style and color of the symbols used in the grid on the canvas, or choose not to display a grid. Workflow elements snap to the grid when they are added or moved around.

**View mode** Settings under the View tab are particularly useful when working with large workflows, as they can be used to remove aspects of the design that are not of immediate interest.

- **Collapsed** Enable this to hide the input and output channels of workflow elements (figure 11.25).
Figure 11.25: The same workflow as above but with the "Collapse" option in the View mode settings enabled.

- **Highlight used elements.** Enabling this option results in elements without at least one input and one output connection to appear faded. Elements connected to those missing connections are also faded (figure 11.26). (Shortcut: Alt + Shift + U)
Figure 11.26: A similar workflow to those above but with the "Highlight used elements" option in the View mode settings enabled. The faded coloring makes it easy to spot that the workflow arm starting with Differential Expression for RNA-Seq is not connected to the rest of the workflow.

- **Rulers** Adds rules along the left vertical and top horizontal edges of the canvas.
- **Auto Layout** Enable this option to adjust the layout automatically every time an element is added and connected. Depending on the workflow design, using the "Layout" option in the right-click menu over the canvas can be preferable (see section 11.1.6).
- **Connections to background** Enable this to put connection lines behind workflow elements (figure 11.27).

See also the Design options, described below, where you can change the color and design of connections.
Figure 11.27: A similar workflow to those above but with the "Connections to background" option in the View mode settings enabled.

**Design** Options under the Design tab allow the shapes and colors of element and connections to be defined. Of particular note is the ability to color elements with non-default configurations differently to those with default settings.

- **Round elements** Round the corners of elements.
- **Show shadow** Add shadows under elements.
- **Coloring** Customize element background colors. Default colors assigned to different types of elements can be specified, and elements with non-default configurations can be assigned a different color to elements with default configurations (figure 11.28). See section 11.2.1 for information on element categories.
- **Connection related options** Customize connection lines between elements, including whether they should be straight or elbow lines, their width and where they should enter and exit elements.
Figure 11.28: A similar workflow to those above, but where standard elements with non-default configuration have been assigned the color pink and control flow elements with non-default configurations have been assigned a pale green color, making them easy to spot.

**Snippets** Snippets are sections of workflows, which have been saved and can be easily added to a new workflow. These are described in section 11.1.8.

### 11.2 Workflow elements

Workflow elements are the building blocks of workflows. Detailed customization of how the workflow will behave can be done by adding and connecting relevant elements, and also by setting individual parameter values, and choosing which parameter values can be edited when launching the workflow. Customization of how the workflow will look when used is also possible by editing workflow element names and the names of their parameter values. These names are used in the wizard generated when the workflow is launched.

In this section, we describe general aspects of workflow elements and then focus on particular element types, the role they play in workflows, and their configuration.

#### 11.2.1 Anatomy of workflow elements

Workflow elements can pass data to other elements, accept data from other elements, or do both of these. Elements that data can enter into and flow out of consist of 3 regions: input channels at the top, output channels at the bottom, and the core section in the middle where the element name is (figure 11.29). A page symbol on the right hand side of the middle section of a workflow element indicates the element can be configured.
Workflow element coloring

Workflow elements are colored according to their category (figure 11.30):

**Light green** Input elements. Elements for taking in the data to be analyzed.

**Dark blue** Output and Export elements. Elements that indicate data should be saved to disk, either in a CLC location (Output elements) or any other accessible file location (Export elements).

**Light grey** An analysis element where the default values are used for all settings.

**Purple** A configured analysis element, i.e. one or more values in that element have been changed from the defaults.

**Emerald green** Control flow elements with default settings.

**Forest green** Configured control flow elements. i.e. one or more values in that element have been changed from the defaults.

Background colors can be changed under the Design tab in the side panel settings of the Workflow editor.

The name of a new element added to a workflow is shown in red text until it is properly connected to other elements.

Configuring Input and Output elements is described in section 11.2.3.

Control flow elements, used to fine tune control of the execution of whole workflows or sections of workflows, are described in section 11.2.4.
11.2.2 Basic configuration of workflow elements

Workflow elements can be renamed, and options within the elements can be customized, affecting what is seen when editing the workflow, what is seen when launching the workflow, and the settings used by default when the workflow is run.

Renaming workflow elements

Renaming workflow elements can aid understanding of a workflow design, especially when there are multiple elements for the same tool. Updated element names are reflected in the workflow launch wizard.

To rename an element, right click on the element name and choose the option “Rename...”, or select the element in the Workflow Editor and press the F2 key.

The Find functionality in the side panel finds elements based the new name.

Both the new name and the original, default name of elements are provided in the Workflow Configuration view (figure 11.31). Terms in the new name or the original, default name can be used in the Filter Elements functionality of this view to find settings for configurable elements. This view is described in section 11.1.7.

Note: The Workflow Configuration view can be opened linked to the Workflow view. Linked views are described further in section 2.1.
Figure 11.31: A workflow before (left) and after (right) the Map Reads to Reference element was renamed. In the linked Workflow Configuration view at the bottom right, both the original and updated element names are listed.

**Configuring options in workflow elements**

Configuring options can be done in a workflow element’s configuration dialog by:

- Double-clicking on an element name in the Workflow view, or
- Right-clicking on an element name and choosing the Configure... option from the menu that appears.

Options can also be edited in the Workflow Configuration view (figure 11.31).

Workflow element customization can include:

1. **Specifying which values can be changed when launching the workflow**

   Beside each option is a lock symbol. Values of unlocked (🔒) options can be changed when launching the workflow. Values of locked (🔒) options cannot be changed when launching the workflow.

   Click on a lock icon to change it from locked (🔒) to unlocked (🔓), or vice versa.

   The values of all locked settings can be seen in the workflow launch wizard, but they are not presented by default (figure 11.33).
Figure 11.32: The Workflow view (top) and Workflow Configuration view (bottom) have been opened as linked views. The Map Reads to Reference element has been opened for configuration in the Workflow view and the Masking mode and Masking track options have been unlocked. They will correspondingly appear unlocked in the Workflow Configuration view after the Finish button is clicked.
**Figure 11.33:** A workflow launch wizard step showing the configurable (unlocked) options at the top, with a heading for the locked settings (top). Clicking on the Locked Settings heading reveals a list of the locked options and their values (bottom)
2. **Changing default values**

When elements are added to a workflow, the default values assigned to options are the same as the defaults for the corresponding tool from the Toolbox. Values for options can be updated in the workflow design. When this is done, the new values become the defaults used when the workflow is run.

3. **Changing option names**

When an option name is changed, the new name is used in the workflow launch wizard. Renaming options can be useful when only a few options can be configured when launching the workflow and their default names are quite generic.

Click on the edit icon (�建) beside an option to edit the name (figure 11.34).

![Figure 11.34: An option originally called "Match score" has been renamed "Score for each match position" in the element configuration dialog. It has also been unlocked so the value for this option will be configurable when launching the workflow.](image)

Note: Clicking on the **Reset** button in a workflow element configuration dialog will reset all changes in that particular configuration step to the defaults, including any updated option names.

### 11.2.3 Configuring input and output elements

**Configuring Workflow Input elements**

Workflow Input elements are the main element type for bringing data into a workflow. At least one such element must be present in a workflow. By default, when a workflow is launched, the workflow wizard will prompt for data to be selected from the Navigation Area, or for data files to be imported on-the-fly using any compatible importer.

Workflow Input elements support the input of CLC format data, as well as raw trace files data. When launching the workflow, data outside CLC locations is selected by choosing the "Select files for import" option. Doing this is referred to as on-the-fly import.

Like other elements, workflow Input elements can be configured to restrict the options available.
for configuration when launching the workflow. See section 11.2.2 for more on locking and unlocking element options.

Common configurations for workflow Input elements include:

- **Configuring import options for primary inputs** Enable or disable selection of input data from the Navigation Area (already imported data) or import of raw data using on-the-fly import. When on-the-fly import is enabled, you can choose whether to restrict the available importers and import settings:
  
  - **Allow any compatible importer** When selected, all compatible importers are displayed as options when launching the workflow. All parameters of these importers will be unlocked, and thus will be available to configure when launching the workflow.
  
  - **Allow selected importers** When selected, one or more importers can be specified. Click on the **Configure Parameters** button to open the "Configure parameters" dialog, where the available import options can be configured for each selected importer. Select the importer to configure from the drop-down list at the top of this dialog. Each option can be locked if desired.

- **Configuring a parameter input with a reference data element** When a Workflow Input element is connected to an input channel expecting reference data, a particular data element can be specified. If that parameter is locked, there will be no opportunity to choose a different element when launching the workflow. If the parameter is left unlocked, the specified element will be used by default, but a different element can be selected when launching.

  Such inputs can also be defined directly in the input channel, rather than connecting an Input element to that channel.


### Configuring Workflow Output and Export elements

Results generated by a workflow are only saved if the relevant output channel of a workflow element is connected to a Workflow Output element or an Export element. Data sent to output channels without an Output or Export element attached are not saved.

Terminal workflow elements with output channels must have at least one Workflow Output element or Export element connected.

The naming pattern for workflow outputs and exports can be specified by configuring Workflow Output elements and Export elements respectively. To do this, double click on a Workflow Output or Export element, or right-click and select the option **Configure...**. Naming patterns can be configured in the **Custom output name** field in the configuration dialog.
The rest of this section is about configuring the Custom output name field, with a focus on the use of placeholders. This information applies to both Workflow Output elements and Export elements. Other configuration settings for Export elements are the same as for export tools, described in section 7.2.2. Placeholders available for export tools, run directly (not via a workflow) are different and are described in section 7.2.3.

**Configuring custom output names**

By default, a placeholders is used to specify the name of an output or exported file, as seen in figure 11.35. Placeholders specify a type of information to include in the output name, and are a convenient way to apply a consistent naming pattern. They are replaced by the relevant information when the output is created.

The placeholders available are listed below. Hover the mouse cursor over the Custom output name field in the configuration dialog to see a tooltip containing this list. Text-based forms of the placeholders are not case specific.

- **{name}** or **{1}** - default name for the tool’s output
- **{input}** or **{2}** - the name of the first workflow input (and not the input to a particular tool within a workflow).
  
  For workflows containing control flow elements, the more specific form of placeholder, described in the point below, is highly recommended.

- **{input:N}** or **{2:N}** - the name of the Nth input to the workflow. E.g. **{2:1}** specifies the first input to the workflow, while **{2:2}** specifies the second input.
  
  Multiple input names can be specified. For example **{2:1}-{2:2}** would provide a concatenation of the names of the first first inputs.

  See section 11.1.3 for information about workflow input order, and section 11.2.4 for information about control flow elements.

- **{metadata}** or **{3}** - the batch unit identifier for workflows executed in batch mode. Depending on how the workflow was configured at launch, this value may be be obtained from metadata. For workflows not executed in batch mode or without Iterate elements, the value will be identical to that substituted using **{input}** or **{2}**.

  For workflows containing control flow elements, the more specific form of placeholder, described in the point below, is highly recommended.

- **{metadata:columnname}** or **{3:columnname}** - the value for the batch unit in the column named “columnname” of the metadata selected when launching the workflow. Pertinent for workflows executed in batch mode or workflows that contain Iterate elements. If a column of this name is not found, or a metadata table was not provided when launching the workflow, then the value will be identical to that substituted using **{input}** or **{2}**.

- **{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.
You can choose any combination of the placeholders and text, including punctuation, when configuring output or export names. For example, \{input\}({day}-{month}-{year}), or \{2\} variant track as shown in figure 11.36. In the latter case, if the first workflow input was named Sample 1, the name of the output generated would be “Sample 1 variant track”.

Figure 11.35: The names that outputs are given can configured. The default naming uses the placeholder \{1\}, which is a synonym for the placeholder \{name\}.

Figure 11.36: Providing a custom name for the result.

It is also possible to save workflow outputs and exports 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 or exports, 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.

**Note:** In some circumstances, outputs from workflow element output channels without a Workflow Output element or an Export element connected may be generated during a workflow run. Such intermediate results are normally deleted automatically after the workflow run completes. If a problem arises such that the workflow does not complete normally, intermediate results may not be deleted and will be in a folder named after the workflow with the word “intermediate” in its name.
11.2.4  Control flow elements

Control flow elements control the flow of data through a workflow. They can be found in the Control Flow folder of the Add Element wizard, as shown in figure 11.37.

![Add Elements](image)

Figure 11.37: Control flow elements are found under the Control Flow folder in the Add Element wizard.

The control flow elements described in this section are:

1. **Iterate**  Used to define a branch of a workflow that should be run multiple times, by splitting its inputs into groups (iteration units, sometimes referred to as batch units).

2. **Collect and Distribute**  Used downstream of an Iterate element to collect all results of an iteration, and group them for collective analysis by further tools.

Further details about each element are provided below.

Note: Like other elements, control flow elements in a workflow can be renamed (see section 11.2.2).

**Iterate**

Elements downstream of an Iteration element are run once for each input, or group of inputs, provided. The inputs to be included in a given iteration are referred to as "batch units" or "iteration units". Multiple Iterate elements can be included in a single workflow.

For workflows with containing a single Input element (green box) and a single Iterate element, batch units can be defined based on the location of the input data or based on information in a metadata table. For any workflow containing multiple Iterate elements, or where there is a single Iterate element and the Batch button is checked when starting the workflow, batch units must be defined using information in a metadata table (section 11.3.2).

Running a workflow with a single Iterate element at the top of a workflow, no downstream Collect and Distribute element (described in the next section), and a single Input element is equivalent to running a similar workflow design without the Iterate element in Batch mode. In this case, the simpler workflow design, without the Iterate element, is usually preferable. In both cases, batch units are set up as described in section 9.3.
CHAPTER 11. WORKFLOWS

Configuring an Iterate element  The configuration options available for an Iterate element are shown in figure 11.38. They are:

1. **Number of coupled inputs** The number of separate inputs for each given iteration. These inputs are "coupled" in the sense that, for a given iteration, particular inputs are used together. For example, when sets of sample reads should be mapped in the same way, but each set should be mapped to a particular reference (figure 11.39).

2. **Error handling** Specify what should happen if an error is encountered. The default is that the workflow should stop on any error. The alternative is to continue running the workflow if possible, potentially allowing later batches to be analyzed even if an earlier one fails.

3. **Metadata table columns** If the workflow is always run with metadata tables that have the same column structure, then it can be useful to set the value of the column titles here, so the workflow wizard will preselect them. The column titles must be specified in the same order as shown in the workflow wizard when running the workflow. Locking this parameter to a fixed value (i.e. not blank) will require the definition of batch units to be based on metadata. Locking this parameter to a blank value requires the definition of batch units to be based on the organization of input data (and not metadata).

4. **Primary input** If the number of coupled inputs is two or more, then the primary input (used to define the batch units) can be configured using this parameter.

![Image of Iterate element configuration](image)

Figure 11.38: The number of coupled inputs in this simple example is 2, allowing each set of sample reads to be mapped to a particular reference, rather than using the same reference for all iterations.

**Collect and Distribute**

Collect and Distribute elements are relevant in workflows with upstream Iterate elements. The steps between an Iterate element and a Collect and Distribute element are referred to as an "iteration block. When a Collect and Distribute element is encountered, intermediate results with that iteration block are collected. That data is passed to downstream elements according to configuration carried out when launching the workflow.
Figure 11.39: Reads can be mapped to specified contigs due to the 2 input channels of the Iterate element. Using this design, a single sequence list containing all the unmapped reads from all the initial inputs is generated. That would not be possible without the inclusion of the Iterate and Collect and Distribute elements.

Configuring a Collect and Distribute element

By default, a Collect and Distribute element has one output channel. In this case, all results from the iteration block are collected and then passed on to downstream steps of the workflow.

More than one output channel can be configured by entering terms in a comma separated list in the Outputs field (figure 11.40). The number of terms determines the number of output channels. Connections between these output channels and input channels of downstream elements determine how data should be distributed in the following stage of the workflow.

If the Collect and Distribute element has more than one output channel, the path taken by a given element is determined by the value in the metadata column specified when launching the workflow. This column can be preconfigured in the Group by metadata column setting.

For example, when launching the workflow in figure 11.41, a metadata column called "Type" was specified for defining which samples were cases and which were controls. The iteration units were defined by the contents of the "ID" column (figure 11.42).
Figure 11.40: A comma separated list of terms in the Outputs field of the Collect and Distribute element defines the number of output channels and their names.

Figure 11.41: In this workflow, each case sample is analyzed against all of the control samples.
Figure 11.42: Contents of the metadata column "Type" define which samples are cases and which are controls. Iteration units are defined by the contents of the "ID" column.
11.2.5 Track lists as workflow outputs

Track lists can be made and saved in workflows by doing the following (see figure 11.43:

- Add a Track List element to the workflow.
- Connect an Output or Export element to the Track List element.
- If you wish to include tracks that are available from within the workflow, add Input elements for each of these (here A, B and C) and connect them to the Tracks input channel. Any tracks based on a compatible genome can be added to a given track list.
- To include tracks generated within the workflow, connect the Output channels of elements producing these tracks to the Tracks input channel of the Track List element (in this example, the Reads Track generated by the Map Reads to Reference element).
- Remember to save all tracks generated by the workflow and used in the Track List: connect an Output element to each output channel connected to the Tracks input channel. This is needed because to view a track list, all the tracks referred to by it must be saved to a CLC data area. When this is not the case, a warning message "Track List: all tracks must also be workflow outputs" is displayed, and the Run and Create Installer buttons are disabled.
- Optionally, configure the order the tracks should be taken into the Track List element. This order is the order the tracks will be shown in the track list that is generated (see section 11.1.2). Note that if a track list contains a variant track, then when the track list is opened, the variant table opens in a split view with the track list. If a track list contains several variant tracks, then the one highest in the list is the one that will have its table opened in the split view.

11.2.6 Input modifying tools

A few tools in the CLC Workbench are input modifying, meaning that when the tool is run from the Toolbox, it directly manipulates the input provided, instead of generating a new element.
as output. However, when such tools are used in a workflow context, they do generate a new element as output\textsuperscript{1}.

Examples of such tools are:

- Find Open Reading Frames
- Gaussian Statistical Analysis
- Pfam Domain Search
- Proportion-based Statistical Analysis

### 11.3 Launching workflows individually and in batches

When a workflow is launched, a launch wizard appears, allowing you to configure available options. Workflows can be launched by:

- Clicking on the Run Workflow button at the bottom a workflow open in the workflow editor.
- Selecting a workflow listed in the Installed Workflows folder in the Workbench Toolbox. Workflows installed in your Workbench are listed in this folder. If you are connected to a CLC Server with installed workflows you have access to, those workflows are also listed here.

Launching installed workflows is done the same way as launching tools:

- Select the workflow from the Toolbox menu at the top of the Workbench (as mentioned above).
- Double-click on the workflow name in the listing in the Toolbox tab in the bottom left side of the Workbench.
- Use the Quick Launch tool, described in section 9.1.

#### Workflow inputs

Inputs can be CLC data elements, selected from the Navigation Area, or files located on any accessible system. This includes CLC Server import/export directories when you are connected to a CLC Server, and files on AWS S3 if you have configured an AWS Connection.

To select files from a non-CLC location, choose the "Select files for import" option in the launch wizard (figure 11.44). When you do this, the files will be imported on the fly, as the first action taken when the workflow runs.

Further details about workflow inputs are provided in section 11.2.3. Details about connecting to other systems are in chapter 6.

\textsuperscript{1}Prior to version 22.0, input modifying tools behaved the same when run from the Toolbox or in a workflow context. Workflow elements for such tools were marked in the Input channel and affected output channel by an M in a circle. There were restrictions for such tools in workflows in these older versions. Please see the manual for the version you are running for full details.
Workflow outputs

Output and Export elements in workflows specify the analysis results to be saved. In addition to analysis results, a Workflow Result Metadata table can be output. This contains a record of the workflow outputs, as described in section 11.3.1.

Launching batches of analyses

When multiple inputs are provided, you can choose to run the workflow multiple times, one time for each input, or for defined sets of inputs that should be analyzed together. This is referred to as running in Batch mode, with the inputs to be analyzed together being referred to as batch units. This is described in section 11.3.2.

In addition, using Iterate and Collect and Distribute control flow elements within a workflow allows for part of a workflow being run once per batch unit, while other parts are run on all the data together. This is described in section 11.3.3.

11.3.1 Workflow Result Metadata tables

A Workflow Result Metadata table can be output as a record of the outputs generated by the workflow. There is a row describing each output, and each data element generated has an association to the relevant metadata table row.

Creating a Workflow Result Metadata table is enabled by default, as shown in figure 11.45. See section 10.3.1 for information on finding and working with data associated with metadata rows.

Workflow Result Metadata tables from batch runs

Workflow Result Metadata tables can be useful when finding results of workflows run in Batch mode due to the large number of outputs that can be produced.
CHAPTER 11. WORKFLOWS

Figure 11.45: The final step when launching a workflow includes an option to create a workflow result metadata table. This is enabled by default.

A single table is generated per batch run, containing information about all the results. A Batch identifier column is included for any outputs that were generated as part of a batch run. This includes outputs from workflows run in Batch mode, and also outputs from steps in an iteration loop, i.e. downstream of an Iterate control flow element (figure 11.46).

Where batch units are defined using metadata, the Workflow Result Metadata table includes original metadata information, where possible.

See section 11.3.2 for more information on launching workflows in batch mode.

See section 11.2.4 for details about control flow elements and section 11.3.3 for more information on launching workflows containing control flow elements.

Figure 11.46: The Workflow Result Metadata table, top left, was generated from a run of the workflow on the right. Here, 4 RNA-Seq Analysis runs occurred within the iteration loop (between the Iterate and the Collect and Distribute elements). Those results were then supplied to Differential Expression in Two Groups, which was run once. There are thus 5 rows in the Workflow Metadata Result table. The RNA-Seq Analysis results each have a batch identifier, while the statistical comparison output does not.

There is one exception to this. Where batch units have been defined by the organization of the input data and the outputs are to be saved in the same folders as the inputs, one workflow result metadata table is generated per analysis.
11.3.2 Running workflows in batch mode

Running batched analyses occurs when:

- The Batch checkbox at the bottom of input steps in the launch wizard has been checked, and/or
- The workflow contains one or more Iterate control flow elements. Steps downstream of Iterate elements and upstream of Collect and Distribute elements, if present, are run once for each batch unit.

When running batches of analyses, the grouping of the data into batch units must be defined after the inputs have been selected.

Defining batch units based on the organization of the input data

For simple workflows, the data to be analyzed together can be defined by its organization. This is identical to defining batch units when launching a tool, as described in section 9.3.

Here "simple workflows" means workflows with just one analysis input that changes per batch, for example, the sets of sequencing reads to be mapped, where the same reference sequence is used for every mapping.

Defining batch units based on metadata

For more complex scenarios, such as when more than one workflow input uses different data in each batch run, or where part of the workflow is run multiple times, and other parts run just once, batch units must be defined using metadata. Metadata can also be used to define batch units for simple workflows.

There are two ways metadata can be provided:

1. Select a CLC Metadata Table as the metadata source. The data elements selected from the Navigation Area as inputs must already have associations to this CLC Metadata Table. If a CLC Metadata Table with data associated to it has been selected in the "Select Workflow Input" step of a workflow, that table will be pre-selected in the "Configure batching" step of the launch wizard. You can specify the column that batch units will be based on. Data associated with the table rows for each unique value in that column make up the contents of the batch units. The contents can be refined using the fields below the preview pane (figure 11.47).

2. Select an Excel, CSV or TSV format file as the source. The metadata in that file is imported into the CLC software at the start of the workflow run. The first column must contain either the exact names of the data selected or a unique prefix, that is, at least enough of the first part of the name to identify each data item uniquely (figure 11.48).

   If data is being imported on the fly, the full file name can include file extensions, but not the path to the data.

   The top row of the file must contain column headers.
Figure 11.47: A CLC Metadata Table with data associated to it was selected as input to a workflow being launched in Batch mode. In the Configure batching wizard step, the metadata source is pre-configured. The column to base batch units on can be selected (top). The Batch overview step shows the data elements in each batch unit. Here "trim" has been entered in the "Only use elements containing" field, so only elements containing the term "trim" in their names are included in the batch units (bottom).

For example, if a data element selected in the Navigation Area has the name atp8a_1_sample1_day3, then the first column could contain that name in full, or just enough of the first part of the name to uniquely identify it. This could be, for example, atp8a_1_sample1. Similarly, if a data file selected for on-the-fly import is at: C:\Users\username\My Data\atp8a_1_sample1_day3clc, the first column of the Excel spreadsheet could contain atp8a_1_sample1_day3clc, or a prefix long enough to uniquely identify the file, e.g. atp8a_1_sample1.

Example: On-the-fly import of single end reads based on metadata

In figure 11.48, a workflow with a single input is being launched in batch mode. The eight files selected contain Illumina single end reads. This raw data will be imported on the fly using metadata to define the batch units. The metadata column in the Excel file that contains information defining the batch units has been specified. Here, files with the same value in the SRR_ID column will be imported and analyzed together.

Each row in the SRR_ID column has a unique entry, so 8 batch units will be made, with one sequence file in each batch unit. If a column containing fewer unique values was selected, one or more batch units would consist of several files. This is illustrated in figure 11.48.

In the next step, a preview of the batch units is shown. The workflow will be run once for each entry in the left hand column, with the input data grouped as shown in the right hand column (figure 11.49).
Example: On-the-fly import of paired end reads based on metadata

When importing data on-the-fly and organizing batch units based on metadata, the metadata must have a row per file being imported. For paired data, this means at least 2 rows per sample. One column in the file must contain information about which files belong together. This allows the sequence list created to be associated with the relevant information.

The contents of an Excel file with information about 2 sets of paired files containing Sanger data are shown in figure 11.50.

Each row for data that is in the same batch unit must contain the same descriptive information. Where there is conflicting information for a given batch unit, the value for that column will be ignored. If all entries for a given column are conflicting, the column will not appear in the resulting CLC Metadata Table.

Saving results from workflows run in batch mode

When a workflow is run in batch mode, options are presented in the last step of the wizard for specifying where to save results of individual batches (see section 9.3).

If the workflow contains Export elements, an additional option is presented, Export to separate directories per batch unit (figure 11.51). When that option is checked, files exported are placed into separate subfolders under the the export folder selected for each export step.

11.3.3 Running part of a workflow multiple times

To run a part of the workflow multiple times, once for each batch unit, add an Iterate control flow element to the workflow. All elements downstream of an Iterate element are run on each batch unit individually, until a Collect and Distribute element is encountered. The parts of the workflow
Figure 11.49: The Batch overview step allows you to review the batch units. In the top image, a column called SRR_ID had been selected, resulting in 8 batch units, so 8 workflow runs, with the data from one input file to be used in each batch. In the lower image, a different column was selected to define the batch units. There, the workflow would be run 3 times with the input data grouped as shown.

Figure 11.50: An Excel file at the top describes 4 Sanger files that make up two pairs of reads. The "Sample Name" column was identified as the one indicating the group the file belongs to. Information about the relevant sample appears in each row. At the Batch overview step, shown at the bottom, you can check the batch units are as intended.

downstream of the Collect and Distribute element are run on all the data together, or in subsets of the data, as configured in the Collect and Distribute element (see section 11.2.4).

For example, the workflow in figure 11.52 would run an RNA-Seq Analysis for each sample separately, and then create a single combined report for the set of samples.

When running on a CLC Server the iterating parts of the workflow is run as separate jobs. For parallel execution of these iterations, job nodes or grid nodes must be available.

When using metadata table to specify the batch units, you are prompted to specify the column
Figure 11.51: Options are presented in the final wizard step for configuring where outputs and exported files from each batch run should be saved.

Figure 11.52: The RNA-Seq analysis tool is run once per sample and a single combined report is then generated for the full set of samples.

that defines how the samples should be grouped for execution. In figure 11.53, grouping by the column "ID" results in the RNA-Seq Analysis tool being run 8 times, once for each sample. Selecting the "Gender" column instead results in the RNA-Seq Analysis tool being run 2 times, once for each value in that column, male and female. The Combine Reports tool runs once, using information from all samples.

The name of workflow elements can be changed. This changes the text displayed in the wizard when the workflow is run. This can be useful if a workflow contains multiple identical elements (figure 11.54).
11.4 Advanced workflow batching

Fine-tuned control of the execution of whole workflows or sections of workflows can be achieved using metadata describing the relationships between particular samples and using control flow elements in a workflow design. Complex analysis goals that can be met in a straightforward manner include:

- **Grouping the data into different subsets to be analyzed together in particular sections of a workflow.** Groupings of data can be used in the following ways:
  - Different groupings of data are used as inputs to different sections of the same workflow. For details, see section 11.3.3 and section 11.4.2.
  - Different workflow inputs follow different paths through parts of a workflow. Based on metadata, samples can be distributed into groups to follow different analysis paths in some workflow sections, at the same time as processing them individually and identically through other sections of the same workflow. Configuring Collect and Distribute elements is central to the design of this workflow. This is described in section 11.2.4. Running such workflows is described in section 11.3.3.

- **Matching particular workflow inputs for each workflow run.** Where more than one input to a workflow changes per run, the particular input data to use for each run can be defined using metadata. The simplest case is as described in section 11.4.1. However, more complex scenarios, such as when intermediate results should be merged or parts of the workflow should be run multiple times, can also be catered for, as described in section ??.
Examples in this section make reference to CLC Genomics Workbench tools and data types commonly analyzed using that software. However, the principles apply equally to workflows created in the *CLC Main Workbench*.

### 11.4.1 Batching workflows with more than one input changing per run

When a workflow contains multiple input elements (multiple light green boxes),

A Batch checkbox is available in the launch wizard for each input element attached to a main input channel.

Checking that box indicates that the data supplied for that input should change in each batch run.

By contrast, if multiple elements are selected, and the Batch option is not checked, all elements will be treated a single set, to be used in a single analysis run.

Most commonly, one input is changed per run. For example, in a batch analysis involving read mappings, usually each batch unit would include a different set of reads, but the same reference sequence.

However, it is possible to have two or more inputs that are different in each batch unit. For example, an analysis involving a read mapping, where each set of reads should be mapped to a different reference sequence. In cases like this, batch units must be defined using metadata.

Figure 11.55 shown an example where the aim is to do just this. The workflow contains a Map Reads to Contigs element and two workflow input elements, Sample Reads and Reference Sequences. The information to define the batch units is provided by two Excel files, one containing information about the Sample Reads input and the other with information about the Reference Sequences input. The contents of files that would work for this example are shown in figure 11.56.

Of particular note are:

- The first column of file contains the exact file names for all data for that input, across all of the batch runs.
- At least one column in each file has the same name as a column in the other file. That column should contain the information needed to match the input data, in this case, the Sample Reads input data with the relevant Reference Sequences input data for each batch unit.

In the Workflow-level batching section of the launch wizard, the following are specified:

- The primary input. The input that determines the number of times the workflow should be run.
- The column in the metadata for the primary input that specifies the group the data belongs to. Each group makes up a single batch unit.
- The column in both metadata files that together will be used to ensure that the correct data from each workflow input are included together in a given batch run. For example, a given
CHAPTER 11. WORKFLOWS

Figure 11.55: A workflow with 2 inputs, where the Batch checkbox had been checked for both in the relevant launch steps. Metadata is used to define the batch units since the correct inputs must be matched together for each run.

Figure 11.56: Two Excel files containing information about the data for each batch unit for the workflow shown in figure 11.55. With the settings selected there, the number of batch runs will be based on the Sample Reads input, and will equal the number of unique SRR_ID entries in the DrosophilaMultiReference.xlsx file. The correct reference sequence to map to is determined by matching information in the Reference column of each Excel file.

set of sample reads will be mapped to the correct reference sequence. A column with this name must be present in each metadata file or table.

In figure 11.55, Sample Reads is the primary input: We wish to run the workflow once for each sample, which here, is once for each SRR_ID entry. The Reference sequence to use for each of these batch units is defined in a column called Reference, which is present in both the file containing information about the samples and the file containing information about the references.
11.4.2 Multiple levels of batching

Sometimes it can be useful to batch or iterate over multiple levels. For example, suppose we have Illumina data from 4 lanes on the flow cell, such that each sample has 4 associated reads list. We may wish to run QC for Sequencing Reads per reads list, but the RNA-Seq Analysis tool per sample. A workflow like the one drawn in figure 11.57 allows us to do this, by connecting Iterate elements directly to each other. The top-level Iterate element results in a subdivision (grouping) of the data, and the innermost Iterate results in a further subdivision (grouping) of each of those groups. Note that it is not necessary for the workflow to include a Collect and Distribute element.

![Workflow diagram](image)

Figure 11.57: The top-level Iterate element results in a subdivision (grouping) of the data, and the innermost Iterate results in a further subdivision (grouping) of each of those groups.

When running the workflow, only metadata can be used to define the groups, because the workflow contains multiple levels of iterations (figure 11.58).

![Workflow diagram](image)

Figure 11.58: When the workflow contains multiple levels of iterations, only metadata can be used to define the groups.

It is always possible to execute a third level of batching by selecting the Batch checkbox when launching the workflow: this will run the whole workflow, including the inner batching processes, several times with different sets of data.

Control flow elements are described in more detail in section 11.2.4.

11.5 Managing workflows

Workflows can be managed from the Workflow Manager:

- **Utilities | Manage Workflows ( Ribbon)**

or using the "Workflows" button ( Ribbon) in the toolbar and then select "Manage Workflows..." ( Ribbon).
The Workflow Manager (figure 11.59) lists workflows installed on your system under the Installed Workflows tab. Workflows under the Installed Workflows tab can be configured, renamed and uninstalled, as described below.

Workflows provided by QIAGEN are listed under the Template Workflows tab.

Copies of workflows can be made by clicking on the "Open Copy of Workflow" button for the relevant workflow. This opens an editable copy of the workflow in the Workflow Editor. That workflow can then, if desired, be saved as an element in Navigation Area, and installed so it appears under the Installed Workflows folder of the Toolbox.

**Note:** Copies of all workflows in the Workbench Toolbox can also be opened from within the Toolbox on the bottom left side of the Workbench. Right-click on the workflow and choose "Open Copy of Workflow".

![Workflow Manager](image)

Figure 11.59: An installed workflow has been selected in the Workflow Manager. Some actions can be carried out on this workflow, and a preview pane showing the workflow design is open on the right hand side.

**Configure**

Clicking on the **Configure** button for an installed workflow will open a dialog where configurable steps in the workflow are shown (figure 11.60). Settings can be configured, and unlocked settings can be locked if desired.

**Note:** Parameters locked in the original workflow design cannot be unlocked. Those locked using the Configure functionality of the Workbench Manager can be unlocked again later in the same way, if desired.

Parameter locking is described further in section 11.2.2.

Note that parameters requiring the selection of data should only be locked 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, or if the parameter is optional and no data should
be selected. If the workflow is intended to be executed on a CLC Server, it is important to select data that is located on the CLC Server.

**Rename**

Clicking on the **Rename** button for an installed workflow allows you to change the name. The workflow will then be listed with that new name in the "Installed Workflows" folder of the Toolbox. The workflow id will remain the same.

**Uninstall**  Use this button to uninstall an installed workflow.

**Description, Preview and Information**  In the right hand pane of the Workflow Manager, are three tabs.

- **Description** Contains the description that was entered when creating the workflow installer (see section 11.5.2).
- **Preview** Contains a graphical representation of the workflow
- **Information** Contains general information about that workflow, such as the name, id, author, etc. (figure 11.61)

The Information tab contains the following:

- **Workflow build id** The date (day month year) followed by the time (based on a 24 hour time) when the workflow installer was created. If the workflow was installed locally without an installation file being created, the build ID will reflect the time of installation.
- **Workflow name** The name of workflow
Figure 11.61: Key workflow information, entered when the workflow installer was created, is available under the Information tab of the Workflow Manager.

- **Referenced data** If reference data was referred to by the workflow and the option Bundled or Reference was selected when the installer was made, the reference data referred to is listed in this field. See section 11.5.2 for further details about these options.

- **Author email** The email address the workflow author entered when creating the workflow installer.

- **Author homepage** The homepage the workflow author entered when creating the workflow installer.

- **Author organization** The organization the workflow author entered when creating the workflow installer.

- **Author name** The workflow author’s name.

- **Workflow version** The version the workflow author assigned to the workflow when creating the installer.

- **Created using Workbench version** The version of the CLC Workbench used when the workflow installer was created.

### 11.5.1 Updating workflows

After installing a new version of a CLC Workbench, workflows may need to be updated before they can be used. Three situations are described in this section:

- Updating workflows stored in the Navigation area

- Updating installed and template workflows when using an upgraded Workbench in the same major version line
• Updating installed workflows when using software in a higher major version line

"Major version line" refers to the first digit in the version number. For example, versions 10.0.1 and 10.5 are part of the same major release line (10). Version 9.x is part of a different major version line (9).

Updating workflows stored in the Navigation area

When you open a workflow stored in the Navigation Area that needs to be updated, an editor will open listing the tools that need to be updated, along with additional information about the changes to the tools (figure 11.62).

To update the workflow, click on the OK button at the bottom of the editor. The updated workflow can be saved under a new name, leaving the original workflow unaffected.

Updating installed and template workflows when using an upgraded Workbench in the same major version line

When working on an upgraded CLC Workbench in the same major version line, installed and template workflows are updated using the Workflow Manager.

To start the Workflow Manager, go to:

Utilities | Manage Workflows

or click on the "Workflows" button in the toolbar, and select "Manage Workflow..." from the menu that appears.

A red message is displayed for each workflow that needs to be updated. An individual workflow can be updated by selecting it and then clicking on the Update... button. Alternatively, click on the Update All Workflows button to carry out all updates in a single action (figure 11.63).

When you update a workflow through the Workflow Manager, the old version is overwritten.
Figure 11.63: A message in red text indicates a workflow needs to be updated. The Update button can be used to update an individual workflow. Alternatively, update all workflows that need updating by clicking on the Update All Workflows button.

To update a workflow you must have permission to write to the area the workflow is stored in. Usually, you will not need special permissions to do this for workflows you installed. However, to update template workflows, distributed via plugins, the CLC Workbench will usually need to be run as an administrative user.

When one or more installed workflows or template workflows needs to be updated, you are informed when you start up the CLC Workbench. A dialog listing these workflows is presented, prompting you to open the Workflow Manager (figure 11.64).

Figure 11.64: A dialog reporting that an installed workflow needs to be updated to be used on this version of the Workbench.

**Updating installed workflows when using software in a higher major version line**

To update an installed workflow after upgrading to software in a higher major version line, you
need a copy of the older Workbench version, which the installed workflow can be run on, 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 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 11.65).

![Figure 11.65: Open a copy of an installed workflow by right-clicking on its name in the Workbench Toolbox.](image)

Save the copy of the workflow. One way to do this is to drag and drop the tab to the location of your choice in the Navigation Area.

Close the older Workbench and open the new Workbench version. In the new version, open the workflow you just saved. Click on the **OK** button if you are prompted to update the workflow.

After checking that the workflow has been updated correctly, including that any reference data is configured as expected, save the updated 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.

### 11.5.2 Creating a workflow installation file

Installing workflows allows you to provide and use a controlled version of the workflow. Only parameters left unlocked in such workflows can be configured when launching. Installed workflows are available for use from within the Workbench Toolbox.

To create a workflow installer, or to directly install the workflow into the Toolbox, click on the **Installation...** button at the bottom of the workflow editor (keyboard shortcut Shift + Alt + I).

At the end of this process, you will have the option to install the workflow directly to your Workbench or to create an installer file, which can be used to install the workflow on any compatible **CLC Workbench** or **CLC Server**. If you are logged into a **CLC Server** as a user with appropriate permissions, you will also have the option to install the workflow directly on the **CLC Server**.

**Information about the workflow**

In the Create Installer dialog, you are prompted for information about the workflow (figure...
11.66). After the workflow is installed, that information will be visible in the Workflow Manager (figure fig:workflowinfoinmgr).

The workflow name and organization information is required, while filling in other fields is optional.

![Create Installer](image)

**Figure 11.66:** Provide information about the workflow that you are creating an installer for.

- **Author name** The name of the workflow author.
- **Author email** The email address of workflow author.
- **Author homepage** A URL.
- **Organization** (Required) The organization name. This is used as part of the workflow id (section 11.5).
- **Workflow name** (Required) The name of the workflow, as it should appear in the Toolbox after installation. Changing this does not affect the name of the original workflow (as appears in your Navigation Area). This name is also used as part of the workflow id (section 11.5).
- **ID** The workflow id. This is created using information provided in other fields. It cannot be directly edited.
- **Workflow icon** An icon to use for this workflow in the Toolbox once the workflow is installed. Icons use a 16 x 16 pixel gif or png file. If your icon file is larger, it will automatically be resized to fit 16 x 16 pixels.
- **Workflow version** A major and minor version.
- **Workflow description** A textual description of the workflow. After installation, this is shown in the Description tab of the Workflow Manager (section 11.5) and is also shown in a tooltip when the cursor is hovered over the installed workflow in the Workbench Toolbox.
Choosing how reference data inputs should be handled

If data elements were preconfigured as inputs to any workflow element, then the next step when creating the workflow installer is to specify how those elements and associated data should be handled (figure 11.69).

In workflows where no data elements were preconfigured as inputs, this step is not shown.

The data handling options are:

**Ignore** The data elements selected as inputs in the original workflow are not included in the workflow installer.

Input options where *Ignore* is selected should generally be kept unlocked. If locked, the data element referred to must present in the exact relative location used on your system when creating the installer. If the option is locked, and the selected data element is not present in the expected location, an error message is shown in the Workflow Manager when the workflow is installed. It will not be possible to run that workflow until the relevant data element is present in the expected location.


**Bundle** The data elements selected as inputs in the original workflow are included in the workflow
installer. This is a good choice when sharing the workflow with others who may not have the relevant reference data on their system.

When installing a workflow with bundled data on a CLC Workbench, you are prompted where to save the bundled data elements. If the workflow is on a CLC Server, the data elements are saved automatically, as described in the CLC Server manual at:


Bundling is intended for use with small reference data elements. With larger elements, the workflow installer can quickly become very big. In addition, each time such a workflow is installed, a copy of the bundled data is included, even if the relevant data element is already present on the system elsewhere.

When working with large data elements, leaving the input option unlocked and choosing the Ignore option is recommended. In this case, the relevant data elements should be shared using other means. For example, export the data and share this separately. The benefit with this, over bundling, is that the data can be shared once, rather than with every workflow installer that refers to it.

![Figure 11.69: Bundling data with the workflow installer.](image)

**Installation options**

The final step asks you to indicate whether to install the workflow directly to your Workbench or to create an installer file, which can be used to install the workflow on any compatible CLC Main Workbench or CLC Server (figure 11.70). If you are logged into a CLC Server as a user with appropriate permissions, you will also have the option to install the workflow directly on the CLC Server.

Workflows installed on a CLC Workbench cannot be updated in place. To update such a workflow, make a copy, modify it, and then create a new installer. We recommend that you increase the version number when creating the installer to help track your changes.

When you then install the updated copy of the workflow, a dialog will pop up with the message "Workflow is already installed" (figure 11.71). You have the option to force the installation. This will uninstall the existing workflow and install the modified version of the workflow. If you choose to do this, the configuration of the original workflow will be gone.
11.5.3 Installing a workflow

Workflow .cpw files can be installed on a CLC Workbench using the Workflow Manager:

Utilities | Manage Workflows (Utilities | Manage Workflows)

or press the "Workflows" button (Workflows) in the toolbar and then select "Manage Workflows..." (Manage Workflows...).

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 11.72).

Figure 11.72: Workflows available in the workflow manager. The alert on the "Variant detection" workflow means that this workflow needs to be updated.
See section 11.5.2 for information about options for handling reference data inputs.

Information about installing workflows on a CLC Server is provided in the CLC Server manual at:

Part III

Bioinformatics
Chapter 12

Viewing and editing sequences

12.1 Sequence Lists

12.1.1 Creating sequence lists

12.1.2 Graphical view of sequence lists

12.1.3 Table view of sequence lists

12.1.4 Annotation Table view of sequence lists

12.1.5 Working with paired sequences in lists

12.2 View sequences

12.2.1 Sequence settings in Side Panel

12.2.2 Selecting parts of the sequence

12.2.3 Editing the sequence

12.2.4 Sequence region types

12.2.5 Circular DNA

12.3 Working with annotations

12.3.1 Viewing annotations

12.3.2 Adding annotations

12.3.3 Editing annotations

12.3.4 Export annotations to a gff3 format file

12.3.5 Removing annotations

12.4 Element information

12.5 View as text

Sequence information is stored in sequence elements or sequence lists. This chapter describes basic functionality for creating an working with these types of elements. Most functionality available for sequence elements is also available for sequence lists.

When you import multiple sequences, they are generally put into a sequence list, and this is the element type in use for most types of work. See chapter 7 for information on importing data.
12.1 Sequence Lists

Sequence list elements contain one or more nucleotide sequences or one or more peptide sequences. They are used as input to many tools, and are generated as output by many tools. Sequence lists can contain single end sequences or paired end sequences, but not a mixture of both. Handling paired data is described at the end of this section.

When open in the viewing area, there are 5 icons at the bottom providing access to different views of the data. Side panel settings on views allow customization of what is shown. These settings can be saved, as described in section 4.6.

In figure 12.1, the graphical view and Table view of a sequence list are open in linked views (section 2.1). With linked views, clicking in one view can lead to updates in the other view. For example, selecting a row in the table view will cause the graphical view to update so the focus is on that sequence.

Much functionality is contained in right-click menus, with options differing depending on which view is open and where you click in it. Some actions are unique to a particular view, while others are available from more than one view. For example, sequence attributes can only be edited in the Table view but sequences can be added from the graphical view and Table view.

![Two views of a sequence list are open in linked views, graphical at the top, and tabular at the bottom. Each view can be customized individually using settings in its side panel on the right.](image)

12.1.1 Creating sequence lists

Sequence lists are created in various ways, including:

- When sequences are imported (chapter 7).
- As outputs of analysis tools.
- When sequences are downloaded, for example using tools under the Download menu.
- Using Create Sequence List, found at:

  Toolbox | Utility Tools ( ) | Sequence Lists ( ) | Create Sequence List ( )
or launched via a button at the bottom sequence lists in Table view. This tool can also be accessed at File | New | Sequence List.

12.1.2 Graphical view of sequence lists

In the graphical view of sequence lists, you can see sequences at the residue level, or zoomed right out to get an overview of the full sequence. Options in the Side Panel on the right allow detailed customization, including aspects like viewing trace data and quality scores, where available (figure 12.1).

Much of the functionality available when working with sequence lists is also available when working with individual sequence elements. Shared functionality is described in section 12.2. This section describes options relevant only to sequence lists.

Menus available when you right-click give access to much functionality for manipulating sequence lists.

Right-click on a blank area to access the following options (figure 12.2):

- **Extract Sequences** Extracts all sequences from the sequence list. If your aim is to extract a subset of the sequences, this can be done from the Table view (see section 12.1.3) or using Split Sequence List view (see section 25.4).
- **Add Sequences** Add sequences from sequence elements or sequence lists to this sequence list.
- **Delete All Annotations from All Sequences** Deleting all annotations on all sequences can be done with this option for sequence lists with 1000 or fewer sequences. In other cases, or for more control over the annotations to delete, use the Annotation Table view, described further below.

Right-click on a sequence or sequence name to access options (figure 12.3) to:

- **Rename, select, open, or delete** that sequence.
- **Sort the sequence list** alphabetically by sequence name, by length or by marked status. These options are only available for sequence lists with 1000 or fewer sequences.
- **Delete sequences that have been marked** This option is enabled when at least one sequence has been marked. Marking sequences is described below.

Tips for working with larger sequence lists are given later in this section.

Marking sequences:

Sequences in a list can be marked. Once marked, those sequences can be deleted, or the sequence list can be sorted based on whether sequences are marked or not. It is easy to adjust markings on many sequences using the options in the right-click menu on selection boxes (figure 12.4).

To mark sequences:
1. Check the **Show selection boxes** option in the "Sequence List Settings" section of the side panel settings on the right hand side. This makes checkboxes visible to the right of each sequence name.

2. Click in the checkbox beside a sequence name to mark the sequence. Clicking in the box for a marked sequence will remove the mark.

![Figure 12.2: Options to extract the sequences in the list, add sequences to the list, and to delete all annotations on all sequences are available when you right-click on a blank area of the graphical view of a sequence list.](image)

**Tips when working with large sequence lists**

- **Sorting long lists** can be done in Table view. For example, to sort on length, ensure the Size column is enabled in the side panel to the right of the table, and then click on the Size column header to sort the list. If a Table view is open as a linked view with the graphical view, clicking on a row in the table will highlight that sequence in the graphical view. See section 2.1 for information on linked views and section 3.2 for information about working with tables.

- **Deleting annotations on sequences** can be done in the Annotation Table view. Right click anywhere in this view to reveal a menu containing relevant options. To delete all annotations on the sequence list, ensure all annotation types are enabled in the side panel settings to the right.

- **To delete many sequences** from a list, you can mark the few you wish to retain, and then invert the marking by right-clicking on any selection checkbox and choosing the option **Invert All Marks** (figure 12.4).

Then right-click on any sequence or sequence name and choose the option to **Delete Marked Sequences** (figure 12.3). If the sequence list contains more than 1000 sequences, a warning will appear noting that, if you proceed, the deletion cannot be undone.
CHAPTER 12. VIEWING AND EDITING SEQUENCES

12.1.3 Table view of sequence lists

Sequence attributes are listed the Table (_excel) view (figure 12.5). This includes information like the name, description, accession, length, etc. of each sequence, as well as any other attributes for that element, such as those added using Update Sequence Attributes in Lists (section 25.3), or custom attributes for that location (section 3.3).

The number of rows reported at the top is the number of sequences in the list (figure 12.5).

Standard functionality for working with tables applies (see section 3.2).

Actions on the sequence list can be taken directly, or via a right-click menu (figure 12.5):

- **Renaming multiple sequences** in a list following the same renaming pattern can be done using the dedicated tool, **Rename Sequences in Lists**, described in section 25.6.
Figure 12.5: In Table view there is a row for each sequence in the sequence list. The number of rows equates to the number of sequences and is reported at the top left side. Right-click to display a menu with actions. This menu differs slightly depending on which column you click upon.

- **Add sequences** Add sequences to this list by dragging and dropping sequence elements or sequence lists from the Navigation Area into the table. Sequences can also be added from the graphical view using a right-click option, as described earlier in this section.

- **Remove sequences** by selecting the relevant rows and either:
  - Clicking on the Delete (🌿) icon in the top toolbar.
  - Right-clicking and choose the option **Delete Sequences** from the menu (figure 12.5).

Sequences can also be deleted from other views.

- **Copy sequence names** Select the relevant rows, right-click and choose **Copy Sequence Names** from the menu. This list can be used within the Workbench, for example, in table filters with the action "is in list" or "is not in list" to find these names in other elements, or they can be pasted to other programs that accept text lists, such as Excel or text editors.

- **Edit attributes** Right-click in the the cell you wish to edit, and then update the contents of that cell. For example, if you right-click on a cell in the Name column, an option called "Edit Name..." will be in the menu presented (figure 12.5).

If you select multiple rows, you will be able to edit the attribute, with the value you provide being applied to all the selected rows.

Values calculated from the sequence itself cannot be edited directly. E.g. The Size column contains the length of each sequence, and the Start of sequence column contains the first 50 residues.

- **Extract selected sequences** Specified sequences in the list can be extracted:
  - **To a new sequence list** by selecting relevant rows and clicking on the **Create New Sequence List** button. This new list must be saved if you wish to keep it.
  - **To a individual sequence elements** by selecting relevant rows and dragging them into the Navigation Area.
Adding attributes

Attributes (columns in Table view) can be added using the right-click menu option Add Attributes. This is good for small lists and simple changes. You are prompted for an attribute name and a single value. A new column is added to the table with the name you provide, and the value you provided is added for all of the selected rows. This option can also be used to edit contents of an existing column, if desired.

The Update Sequence Attributes in Lists tool supports more detailed work, including importing from external sources, such as Excel and CSV format files. See (section 25.3) for more details.

12.1.4 Annotation Table view of sequence lists

Options in the Annotation Table view are the same as those in this view for sequences. They are described in the following sections:

- Working with the Annotation Table view: section 12.3.1
- Adding annotations: section 12.3.2
- Editing annotations: section 12.3.3
- Exporting annotations to a gff3 format file: section 12.3.4
- Deleting annotations: section 12.3.5

12.1.5 Working with paired sequences in lists

When paired sequences are imported, the sequence list created is marked as containing paired data. This can be seen in the Element info view and can be edited within that view (section 12.4).

A sequence list can contain single end reads or it can contain paired end reads. A given list cannot contain a mixture of these.

To create a paired sequence list from existing sequence lists, for example by merging lists, the input lists must be marked as paired and must have the same distance settings. If the input lists do not meet these criteria, a message is shown warning that the resulting sequence list will be unpaired (figure 12.6).

Figure 12.6: A warning appears when trying to create a new sequence list from a mixture of paired and unpaired sequence lists.
12.2 View sequences

This section describes options available when viewing and editing sequences. Many of these options also apply to sequence lists (section 12.1) and alignments (section 14.2).

12.2.1 Sequence settings in Side Panel

Settings for a view are available in the Side Panel at the right hand side of the view (figure 12.7). The view is instantly updated when these settings are changed. The options available for sequence views are described in the sections that follow.

Note: Side Panel settings are not automatically saved when you save the sequence. See section 4.6 for information on saving and applying view settings.

For general information about Side Panels, see section 2.1.6.

Figure 12.7: Overview of the Side Panel for a sequence. Each tab can be expanded to reveal settings that can be configured.

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 12.3.1.

**Restriction sites**
Please see section 21.1.1.

**Motifs**
See section 16.9.1.

**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](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 [12.2.2](#).
∗ +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 17.4.
− 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.

• Quality scores. For sequencing data containing quality scores, the quality score information can be displayed along the sequence.

− Show as probabilities. Converts quality scores to error probabilities on a 0-1 scale, i.e. not log-transformed.
− Foreground color. Colors the letter using a gradient, where the left side color is used for low quality and the right side color is used for high quality. The sliders just above the gradient color box can be dragged to highlight relevant levels. The colors can be changed by clicking the box. This will show a list of gradients to choose from.
− Background color. Sets a background color of the residues using a gradient in the same way as described above.
− Graph. The quality score is displayed on a graph (Learn how to export the data behind the graph in section 7.4).
  ∗ Height. Specifies the height of the graph.
  ∗ Type. The graph can be displayed as Line plot, Bar plot or as a Color bar.
  ∗ Color box. For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

• Trace data. See section 19.1.

• 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.
CHAPTER 12. VIEWING AND EDITING SEQUENCES

- **Graph.** The G/C content level is displayed on a graph (Learn how to export the data behind the graph in section 7.4).
  
  - **Height.** Specifies the height of the graph.
  - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
  - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

  When zoomed out, the graph displays G/C content only for a subset of evenly spaced positions. Because insertions shifts reference positions, zoomed-out graphs with and without insertions may not be directly comparable, as G/C content may be displayed for different positions.

- **Secondary structure.** Allows you to choose how to display a symbolic representation of the secondary structure along the sequence.

  See section 22.2.3 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 18.3.1.

- **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 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 21.1.1.

**12.2.2 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 non-contiguous parts of a sequence (multiselect) You can select non-contiguous 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 12.2.1.)

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.

12.2.3 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 21.1.3.

Note When editing annotated nucleotide sequences, the annotation content is not updated automatically (but its position is). Please refer to section 12.3.3 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.

### 12.2.4 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 12.8 is an example of three regions with separate colors.

![Figure 12.8: Three regions on a human beta globin DNA sequence (HUMHBB).](image)

Figure 12.9 shows an artificial sequence with all the different kinds of regions.

### 12.2.5 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" (ísticas)

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 12.10.

This view of the sequence shares some of the properties of the linear view of sequences as described in section 12.2, 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.
Figure 12.9: 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.

Figure 12.10: A molecule shown in a circular view.

- **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.

**Using split views to see details of the circular molecule**
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 (⌘F) at the bottom of the view

This will open a linear view of the sequence below the circular view. When you zoom in on the linear view you can see the residues as shown in figure 12.11.

Figure 12.11: Two views showing the same sequence. The bottom view is zoomed in.

Note! If you make a selection in one of the views, the other view will also make the corresponding selection, providing an easy way for you to focus on the same region in both views.

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 12.12).

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 12.13).
CHAPTER 12. VIEWING AND EDITING SEQUENCES

Figure 12.12: Double angle brackets mark the start and end of a circular sequence in linear view (top). The first line in the text view (bottom) contains information that the sequence is circular.

Figure 12.13: Right-click on a circular sequence to move the starting point to the selected position.
12.3 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 13.4.2), and atom groups defined on the structure transferred to sequence annotations or vica versa (section 13.4.3).
- You can manually add annotations to a sequence (described in the section 12.3.2).

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 25.1.

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.

12.3.1 Viewing annotations

Annotations can be viewed in a number of different ways:

- As graphical arrows or boxes in all views displaying sequences (sequence lists, alignments etc)
- In a table in the Annotation view
- In the text view of sequences

These views are described in more detail in the following sections.

View Annotations in sequence views

Figure 12.14 shows an annotation displayed on a sequence.

The various sequence views listed in section 12.3.1 have different default settings for showing annotations. However, they all have two groups in the Side Panel in common:

- Annotation Layout
- Annotation Types
Figure 12.14: An annotation showing a coding region on a genomic DNA sequence.

Figure 12.15: 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 12.15.

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 12.16).

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 12.17) 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.
CHAPTER 12. VIEWING AND EDITING SEQUENCES

Figure 12.16: Browsing the gene annotations on a sequence.

Figure 12.17: Example of a torn annotation on a sequence.

View Annotations in a table

The Annotation Table view lists the annotations in a table. Standard functionality for working with tables applies, as described in section 3.2. Here we highlight functionality of particular interest when working with annotations.

This view is useful for getting a quick overview of annotations, and for filtering so that only the annotations of interest are listed. From this view, you can edit and add annotations, export selected annotations to a gff3 format file, and delete annotations. This functionality is described in more detail below.

To open the Annotation Table view:

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 12.18).

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.
12.3.2 Adding annotations

Annotations can be added to a sequence from the graphical view or the Annotation Table view.

**Graphical view** Select the region you want the annotation to cover, right-click on the selected region, and choose the option Add Annotation ( ) from the menu.

See section 12.2.2 on how to make selections that are not contiguous.

**Annotation Table view** Right-click anywhere in this view and select the option Add Annotation ( ) from the menu. (Note that this option is not enabled in sequence lists.)

In both cases, a dialog opens where you can provide information about the annotation (figure 12.19).

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 1.

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.

---

1Note that your own annotation types will be converted to “unsure” when exporting in GenBank format. As long as you use the sequence in CLC format, you own annotation type will be preserved.
(Whether the name is actually shown depends on the Annotation Layout preferences, see section 12.3.1).

- **Type.** Reflects the left-hand part of the dialog as described above. You can also choose directly in this list or type your own annotation type.

- **Region.** If you have already made a selection, this field will show the positions of the selection. You can modify the region further using the conventions of DDBJ, EMBL and GenBank. The following are examples of how to use the syntax (based on http://www.ncbi.nlm.nih.gov/collab/FT/):
  - **467.** Points to a single residue in the presented sequence.
  - **340..565.** Points to a continuous range of residues bounded by and including the starting and ending residues.
  - **<345..500.** Indicates that the exact lower boundary point of a region is unknown. The location begins at some residue previous to the first residue specified (which is not 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 12.3.1) 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. "digitalinsights.qiagen.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.

### 12.3.3 Editing annotations

Annotations can be edited from the graphical sequence view or from the Annotation Table view.

For general information about the Annotation Table view, see section 12.3.1.

**Basic annotation editing in the sequence view**

To edit an annotation from the sequence view:

**Right-click on the annotation | Edit Annotation**

A dialog like that in figure 12.19 will appear. Edit the fields as needed and click on **OK** to save your changes.

**Editing annotations in the Annotation Table view**

In the Annotation Table view, each part of an annotation can be updated by double-clicking in a cell, editing the contents, and pressing **Enter**. In addition, editing options are among the options available in the right-click menu (figure 12.20).

Options related to editing annotations in this menu are:
CHAPTER 12. VIEWING AND EDITING SEQUENCES

Figure 12.20: The right-click menu in the Annotation Table view contains options for adding, editing, exporting and deleting annotations.

- **Edit Annotation...** This option is only enabled if a single annotation is selected in the table. It will open the same dialog used to edit annotations from the sequence view (figure 12.19).

- **Advanced Rename...** Choose this to rename the selected annotations using qualifiers or annotation types. The options in the Rename dialog (figure 12.21) are:
  - **Use this qualifier** Choose the qualifier to use as that annotation name from a drop-down list of qualifiers available in the selected annotations. Selected annotations that do not include the selected qualifier will not be renamed. If an annotation has multiple qualifiers of the same type, the first is used for renaming.
  - **Use annotation type as name** The annotation’s type will be used for the annotation name. E.g. if you have an annotation of type "Promoter", it will get "Promoter" as its name by using this option.

- **Advanced Retype...** Choose this to edit the type of one or more annotations. The options in the Retype dialog (figure 12.22) are:
  - **Use this qualifier** Choose the qualifier to use as the annotation type from a drop-down list of qualifiers available in the selected annotations. Selected annotations that do not include the selected qualifier will not be retyped. If an annotation has multiple qualifiers of the same type, the first is used for the new type.
  - **New type** Enter an annotation type to apply or click on the arrows at the right of the field to see a drop-down list of pre-defined annotation types.
  - **Use annotation name as type** Use the annotation name as its type. E.g. if you have an annotation named "Promoter", it will get "Promoter" as its type by using this option.

### 12.3.4 Export annotations to a gff3 format file

Annotations selected in the Annotation Table view can be exported to a gff3 format file by selecting the option **Export Selection to GFF3 File** in the right-click menu (figure 12.20).
12.3.5 Removing annotations

Annotations can be hidden using the Annotation Types preferences in the Side Panel to the right of the view (see section 12.3.1). 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
12.4 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 12.23.

Figure 12.23: 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 the History view, described in section 2.5 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.

### 12.5 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 12.4.)

In the **Side Panel**, you find a search field for searching the text in the view.
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.

If you have problems viewing 3D structures, please check your system matches the requirements for 3D viewers. See section 1.3.

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
13.1 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

13.1.1 From the Protein Data Bank

Molecule structures can be imported in the workbench from the Protein Data Bank using the "Download" function:

Toolbar | Download (下载) | Search for PDB structures at NCBI (NCBI搜索)

Type the molecule name or accession number into the search field and click on the "Start search" button (as shown in figure 13.1). The search hits will appear in the table below the search field.

![Figure 13.1: 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.](image)

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.

13.1.2 From your own file system

A PDB file can also be imported from your own file system using the standard import function:

Toolbar | Import (导入) | Standard Import (标准导入)
In the Import dialog, select the structure(s) of interest from a data location and tick "Automatic import" (figure 13.2). 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 13.2: A PDB file can be imported using the Standard Import tool.](image)

### 13.1.3 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 13.3).

![Figure 13.3: 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.](image)

Click Next and choose program and database (figure 13.4). 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 13.4: Select database and program.](image)

Please refer to section 24.1.1 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 13.5). 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**.

### 13.1.4 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 13.6).

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
Figure 13.5: 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 13.6: 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.

13.2 Viewing molecular structures in 3D

An example of a 3D structure that has been opened as a Molecule Project is shown in figure 13.7.

If you have problems viewing 3D structures, please check your system matches the requirements for 3D viewers. See section 1.3.
**CHAPTER 13. 3D MOLECULE VIEWER**

**Moving and rotating** The molecules can be rotated by holding down the left mouse button while moving the mouse. The right mouse button can be used to move the view.

Zooming can be done with the scroll-wheel or by holding down both left and right buttons while moving the mouse up and down.

All molecules in the Molecule Project are listed in categories in the Project Tree. The individual molecules or whole categories can be hidden from the view by un-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 (lander) 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.

**13.3 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.

13.3.1 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 Cα 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 13.3.1).

It is possible to adjust the opacity of a surface by adjusting the transparency slider at the bottom of the menu.
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 13.9).

- 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.

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.

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, see section 13.4.1. A
  split-view will appear with a sequence editor for each of the sequence data types (Protein,
  DNA, RNA) (figure 13.12). 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, see section 13.4.2. 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.

![Image of 3D Molecule Viewer](image)

Figure 13.12: 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.

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 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.
CHAPTER 13. 3D MOLECULE VIEWER

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 13.13). Double-clicking an entry in the Project Tree will have the same effect.

Figure 13.13: The "Fit to screen" button can be used to bring a particular molecule or category of molecules in focus.

13.3.2 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 editor for each of the sequence data types (Protein, DNA, RNA). This is described in section 13.4.1.

- **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 13.4.2.

- **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 13.4.3.
• **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 13.5.

**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.

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:
Figure 13.14: Selecting two, three, or four atoms will display the distance, angle, or dihedral angle, respectively.

- **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 (Graphics button). 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).

### 13.4 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 13.4.3).
13.4.1 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 13.15). 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 13.3.1. 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 13.4.2).

Figure 13.15: Protein chain sequences and DNA sequences are shown in separate views.

13.4.2 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 13.4.1) or Align to Existing Sequence.

Select a single protein chain in the project tree (see figure 13.16). 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.

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, see section 13.4.1), 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 14.1) with the following settings:

- Gap open cost: 10.0
- Gap Extension cost: 1.0
- End gap cost: free
- Existing alignments are not redone
Figure 13.16: Select a single protein chain in the Project Tree and invoke "Align to Existing Sequence".

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 13.4.3). Note 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 14.2

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 13.19 and section 14.3).

13.4.3 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 12.3 13.3.1 and more about atom groups in section 13.3.1.

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 13.4.1 13.4.2) or by mapping to an existing sequence or sequence alignment using the 'Align to Existing Sequence' button (see section 13.4.2).

Invoke the Transfer Annotations dialog by selecting a linked protein chain in the Project Tree and press 'Transfer Annotations' (see figure 13.17).

The dialog contains two tables (see figure 13.18). 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,
Figure 13.17: Select a single protein chain in the Project Tree and invoke "Transfer Annotations".

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.

**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 13.19 and section 14.3).
Figure 13.18: The Transfer Annotations dialog allow you to select annotations listed in the two tables, and copy them from structure to sequence or vice versa.

Figure 13.19: Copy annotations from sequences in the alignment to the sequence linked to the 3D view.

13.5 Align Protein Structure

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 13.20). 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.

The dialog box contains three fields:

- **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.

### 13.5.1 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 13.20. 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 13.21. 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 13.21: 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 13.22). Using the new atom group as the reference in the alignment dialog leads to the alignment shown in figure 13.23. 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 13.23. 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 13.24.
Figure 13.22: Creation of an atom group containing the N-terminal domain of calmodulin.

Figure 13.23: Alignment of the same two calmodulin proteins as in figure 13.21, 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.
Figure 13.24: Alignment of the same two calmodulin domains as in figure 13.21, but this time with a focus on the calcium atom within the black box of figure 13.23. The calcium atoms are less than 1 Å apart – compatible with thermal motion encoded in the atoms’ temperature factors.
13.5.2 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:

$$\text{TM-score} = \frac{1}{L} \sum_{i} \frac{1}{1 + \frac{d_i^2}{d(L)^2}}$$

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 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).

13.6 Generate Biomolecule

Protein structures imported from a PBD file show the tertiary structure of proteins, but not necessarily the biologically relevant form (the quaternary structure). Oftentimes, several copies of a protein chain need to arrange in a multi-subunit complex to form a functioning biomolecule. In some PDB files several copies of a biomolecule are present and in others only one chain from a multi-subunit complex is present. In many cases, PDB files have information about how the molecule structures in the file can form biomolecules.
When a PDB file with biomolecule information available has been either downloaded directly to the workbench using the Search for PDB Structures at NCBI or imported using Import Molecules with 3D Coordinates, the information can be used to generate biomolecule structures in CLC Main Workbench.

The "Generate Biomolecule" dialog is invoked from the Side Panel of a Molecule Project (figure 13.25). The button is found in the Structure tools section below the Project Tree.

Figure 13.25: The Generate Biomolecule dialog lists all possibilities for biomolecules, as given in the PDB files imported to the Molecule Project. In this case, only one biomolecule option is available. The Generate Biomolecule button that invokes the dialog can be seen in the bottom right corner of the picture.

There can be more than one biomolecule description available from the imported PDB files. The biomolecule definitions have either been assigned by the crystallographer solving the protein structure (Author assigned = "Yes") or suggested by a software prediction tool (Author assigned = "No"). The third column lists which protein chains are involved in the biomolecule, and how many copies will be made.

Select the preferred biomolecule definition and click OK.

A new Molecule Project will open containing the molecules involved in the selected biomolecule (example in figure 13.26). If required by the biomolecule definition, copies are made of protein chains and other molecules, and the copies are positioned according to the biomolecule information given in the PDB file. The copies will in that case have "s1", "s2", "s3" etc. at the end of the molecule names seen in the Project Tree.

If the proteins in the Molecule Project already are present in their biomolecule form, the message "The biological unit is already shown" is displayed, when the "Generate Biomolecule" button is clicked.

If the PDB files imported or downloaded to a Molecule Project did not hold biomolecule information, the message "No biological unit is associated with this Molecule Project" is shown, when the Generate Biomolecule button is clicked.
Figure 13.26: One of the biomolecules that can be generated after downloading the PDB 2R9R to CLC Main Workbench. It is a voltage gated potassium channel.
Chapter 14

Sequence alignment

Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.1</td>
<td>Create an alignment</td>
<td>290</td>
</tr>
<tr>
<td>14.1.1</td>
<td>Gap costs</td>
<td>291</td>
</tr>
<tr>
<td>14.1.2</td>
<td>Fast or accurate alignment algorithm</td>
<td>292</td>
</tr>
<tr>
<td>14.1.3</td>
<td>Aligning alignments</td>
<td>293</td>
</tr>
<tr>
<td>14.1.4</td>
<td>Fixpoints</td>
<td>293</td>
</tr>
<tr>
<td>14.2</td>
<td>View alignments</td>
<td>295</td>
</tr>
<tr>
<td>14.2.1</td>
<td>Bioinformatics explained: Sequence logo</td>
<td>298</td>
</tr>
<tr>
<td>14.3</td>
<td>Edit alignments</td>
<td>299</td>
</tr>
<tr>
<td>14.3.1</td>
<td>Realignment</td>
<td>301</td>
</tr>
<tr>
<td>14.4</td>
<td>Join alignments</td>
<td>304</td>
</tr>
<tr>
<td>14.5</td>
<td>Pairwise comparison</td>
<td>305</td>
</tr>
<tr>
<td>14.5.1</td>
<td>The pairwise comparison table</td>
<td>306</td>
</tr>
<tr>
<td>14.5.2</td>
<td>Bioinformatics explained: Multiple alignments</td>
<td>307</td>
</tr>
</tbody>
</table>

CLC Main Workbench can align nucleotides and proteins using a *progressive alignment* algorithm (see section 14.5.2.

This chapter describes how to use the program to align sequences, and alignment algorithms in more general terms.

### 14.1 Create an alignment

Alignments can be created from sequences, sequence lists (see section **12.1**), existing alignments and from any combination of the three.

To create an alignment, go to:

**Toolbox | Alignments and Trees | Create Alignment**

This opens the dialog shown in figure **14.1**.

After selecting the elements to align, you are presented with options that can be configured (figure **14.2**).
14.1.1 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 14.3 and 14.4 illustrate the differences between the different gap scores at the sequence ends.

![Figure 14.3](image)

**Figure 14.3:** 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.

### 14.1.2 Fast or accurate alignment algorithm

**CLC Main Workbench** has two algorithms for calculating alignments:

- **Fast (less accurate).** Use an optimized alignment algorithm that is very fast. This is particularly useful for data sets with very long sequences.
- **Slow (very accurate).** The recommended choice unless the processing time is too long.

Both algorithms use progressive alignment. The faster algorithm builds the initial tree by doing more approximate pairwise alignments than the slower option.
14.1.3 Aligning alignments

The **Redo alignment** option becomes available when an alignment is provided as input to the tool (figure 14.1).

- **Leave this box unchecked** when aligning additional sequences to the original alignment. Equal sized gaps may be inserted in all sequences of the original alignment to accommodate the alignment of the new sequences (figure 14.5), but apart from this, positions in the original alignment are fixed.

- **Check this box** to realign the sequences in the alignment provided as input. This can be useful, for example, if you wish to realign using different gap costs than used originally.

A new alignment is output, whether or not **Redo alignment** option is checked.

14.1.4 Fixpoints

To force particular regions of an alignment to be aligned to each other, there are two steps:

1. Add fixpoint annotations to these regions.
2. Check the "Use fixpoints" option when launching the Create Alignment tool.

To add a fixpoint, open the sequence or alignment and:

Select the region you want to use as a fixpoint | Right-click on the selection | Set Alignment Fixpoint Here
**Figure 14.5:** The original alignment is shown at the top. That alignment and a single additional sequence, with four Xs added for illustrative purposes, were used as input to Create Alignment. The "Redo alignment" option was left unchecked. The resulting alignment is shown at the bottom. Gaps have been added, compared to the original alignment, to accommodate the new sequence. All other positions are aligned as they were in the original alignment.

This will add an annotation of type "Alignment fixpoint", with name "Fixpoint" to the sequence (figure 14.6).

Regions with fixpoint annotations with the same name are aligned to each other. Where there are multiple fixpoints of the same name on sequences, the first fixpoints on each sequence will be aligned to each other, the second on each sequence will be aligned to each other, and so on.

To adjust the name of a fixpoint annotation:

- **Right-click the Fixpoint annotation | Edit Annotation ( ) | Type the name in the 'Name' field**

An example where assigning different names to fixpoints is useful: Given three sequences, A, B and C, where A and B each have one copy of a domain while sequence C has two copies of the domain, you can force sequence A to align to the first copy of the domain in sequence C and sequence B to align to the second copy of the domain in sequence C by naming the fixpoints accordingly. E.g. if the fixpoints in sequence C were named ‘fp1’ and ‘fp2’, the fixpoint in sequence A was named ‘fp1’ and the fixpoint in sequence B was named ‘fp2’, then when these sequences are aligned using fixpoints, the fixpoint in sequence A would be aligned to the first copy of the domain in sequence C, while the fixpoint in sequence B would be aligned to the second copy of the domain in sequence C.

The result of an alignment using fixpoints is shown in figure 14.7.
CHAPTER 14. SEQUENCE ALIGNMENT

Figure 14.6: Select a region and right-click on it to see the option to set a fixpoint. The second sequence in the list already has a Fixpoint annotation.

Figure 14.7: Fixpoints have been added to 2 sequences in an alignment, where the first 3 sequences are very similar to each other and the last 3 sequences are very similar to each other (top). After realigning using just these 2 fixpoints (bottom), the alignment now shows clearly the 2 groups of sequences.

14.2 View alignments

The basic options for viewing alignments are the same as for viewing sequences, described in section 12.2.

Alignment specific view options in the Alignment info and the Nucleotide info side panel tabs, to the right of the view, are described here.

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 7.4.

  - **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 7.4).
  
  - **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 14.2.1 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.
14.2.1 Bioinformatics explained: Sequence logo

In the search for homologous sequences, researchers are often interested in conserved sites/residues or positions in a sequence which tend to differ a lot. Most researches use alignments (see Bioinformatics explained: multiple alignments) for visualization of homology on a given set of either DNA or protein sequences. In proteins, active sites in a given protein family are often highly conserved. Thus, in an alignment these positions (which are not necessarily located in proximity) are fully or nearly fully conserved. On the other hand, antigen binding sites in the F\textsubscript{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 14.8). This is also the case for repressor sites as seen for the Cro repressor of bacteriophage \( \lambda \).

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 14.8). 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 14.8 an un-gapped alignment of 11 \textit{E. coli} start codons including flanking regions are shown. In this example, a consensus sequence would only display ATG as the start codon, 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 \( \log_2 4 = 2 \) bits for DNA/RNA and \( \log_2 20 \approx 4.32 \) bits for proteins.
Figure 14.8: 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]

14.3 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 14.1). 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 14.9).
Note! Residues can only be moved when they are next to a gap.

```
  A G G   G A G T C A T
  A G G   G A G T C A T
  A G G   G A G C A G T
  - - - - - - - - - - -
  A G G   G T A C A G T
  - - -  G A G T A G C
  - - -  G A G T A G C
  - - -  G A G T A G G
  A T G   G T G C A C C
  A T G   G T G C A T C
```

Figure 14.9: 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 | right-click the 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 14.1).

The same procedure can be used for joining two alignments.

### 14.3.1 Realignment

This section describes realigning parts of an existing alignment. To realign an entire alignment, consider using the "Redo alignment" option of the **Create Alignment** tool, described in section 14.1.3
Examples where realigning part of an alignment can be helpful include:

- **Adjusting the number of gaps** If a region has more gaps than is useful, select the region of interest and realign using 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 those residues, and then realign the section of interest using those fixpoints, as described in section 14.1.4. This should result in the two residues being aligned, and everything in the selected region around them being adjusted to accommodate that change.

**Realigning a subsection of an alignment**

There are two steps to realigning a subsection of an alignment:

1. Select the region of interest.
2. Realign the selected region.

**Selecting a region**

Click and drag to select the regions of interest. For small regions in a small number of sequences, this may be easiest while zoomed in fully, such that each residue is visible. For realigning entire sequences, zooming out fully may be helpful.

As selection involves clicking and dragging the mouse, all regions of interest must be contiguous. That is, you must be able to drag over the relevant regions in a single motion. This may mean gathering particular sequences into a block. There are two ways to achieve this:

1. Click on the name of an individual sequence and drag it to the desired location in the alignment. Do this with each relevant sequence until all those of interest are placed as desired.

2. Check the option "Show selection boxes" in the Alignment settings section of the side panel settings (figure 14.10). Click in the checkbox next to the names of the sequences you wish to select. Then right-click on the name of one of the sequences and choose the option "Sort Sequences by Marked Status". This will bring all selected sequences to the top of the alignment.

   If you have many sequences to select, it can be easiest to select the few that are not of interest, and then invert the selection by right-clicking on any of the checkboxes and choosing the option "Invert All Marks".

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 14.11), you can choose the option "Open selection in a new view", with the ability to run any relevant tool on that sub-alignment.

**Realign the selected region**
To realign the selected region:

**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 14.12).

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.
Learn more about the options available to you in the sections 14.1.1 and 14.1.2.

14.4 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 14.13.

![Figure 14.13: Selecting two alignments to be joined.](image)

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 14.14.

![Figure 14.14: Selecting order of concatenation.](image)

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 14.15.

**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
Figure 14.15: 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.

shown in figure 14.15 "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.

14.5 Pairwise comparison

For a given set of aligned sequences it is possible to make a pairwise comparison in which each pair of sequences are compared to each other. This provides an overview of the diversity among the sequences in the alignment.

In CLC Main Workbench this is done by creating a comparison table:

Toolbox | Alignments and Trees ( ) | Create Pairwise Comparison ( )

This opens the dialog displayed in figure 14.16:

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,
CHAPTER 14. SEQUENCE ALIGNMENT

Figure 14.16: Creating a pairwise comparison table.

as shown in figure 14.17.

Figure 14.17: Adjusting parameters for pairwise comparison.

- **Gaps** Calculates the number of alignment positions where one sequence has a gap and the other does not.

- **Identities** Calculates the number of identical alignment positions to overlapping alignment positions between the two sequences. 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.

### 14.5.1 The pairwise comparison table

The table shows the results of selected comparisons (see an example in figure 14.18). 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.

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.

### 14.5.2 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 14.19) 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 14.19: 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.](image)

**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 groups 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 15

Phylogenetic trees

Contents

15.1 K-mer Based Tree Construction ........................................... 312
15.2 Create tree ........................................................................... 313
15.3 Model Testing ......................................................................... 314
15.4 Maximum Likelihood Phylogeny ................................................. 316
  15.4.1 Bioinformatics explained ..................................................... 319
15.5 Tree Settings ........................................................................... 324
  15.5.1 Minimap ........................................................................... 325
  15.5.2 Tree layout ........................................................................ 325
  15.5.3 Node settings ..................................................................... 327
  15.5.4 Label settings .................................................................... 327
  15.5.5 Background settings ............................................................. 329
  15.5.6 Branch layout ................................................................. 330
  15.5.7 Bootstrap settings ............................................................... 330
  15.5.8 Visualizing metadata ........................................................... 331
  15.5.9 Node right click menu .......................................................... 331
15.6 Metadata and phylogenetic trees ............................................. 333
  15.6.1 Table Settings and Filtering .................................................. 335
  15.6.2 Add or modify metadata on a tree ........................................ 335
  15.6.3 Undefined metadata values on a tree .................................... 336
  15.6.4 Selection of specific nodes .................................................. 337

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 15.4.1 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 through 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 14.1) 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 15.4.1). 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 15.4.1). 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 15.4 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 15.3) 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 15.4.1).

### 15.1 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 15.1):

![Figure 15.1: Select sequences needed for creating a tree with K-mer based tree construction.](image)

Next, select the construction method, specify the k-mer length and select a distance measure
for tree construction (figure 15.2):

- **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 15.4.1):
    - 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 15.2: 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 section 15.4.1 for further details.

### 15.2 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 15.3:

Figure 15.3: 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 15.4 shows the parameters that can be set for this distance-based tree creation:

- **Tree construction (see section 15.4.1)**
  - **Tree construction method**
    - The **UPGMA** method. Assumes constant rate of evolution.
    - The **Neighbor Joining** method. Well suited for trees with varying rates of evolution.
– Nucleotide distance measure
  * Jukes-Cantor. Assumes equal base frequencies and equal substitution rates.
  * Kimura 80. Assumes equal base frequencies but distinguishes between transi-
tions 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 15.4.1). 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 15.4.1.

15.3 Model Testing

As the Model Testing tool can help identify the best substitution model (15.4.1) 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 15.5):

![Figure 15.5: Select alignment for model testing.](image)

Specify the parameters to be used for model testing (figure 15.6):

![Figure 15.6: Specify parameters for model testing.](image)

• 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.

### 15.4 Maximum Likelihood Phylogeny

To generate a maximum likelihood based phylogenetic tree, go to:
Toolbox | Alignments and Trees | Maximum Likelihood Phylogeny

First, select the alignment to be used for the reconstruction (figure 15.7).

![Maximum Likelihood Phylogeny](image)

**Figure 15.7: Select the alignment for tree construction.**

You can then set up the following parameters (figure 15.8):

![Adjusting parameters for maximum likelihood phylogeny](image)

**Figure 15.8: Adjusting parameters for maximum likelihood phylogeny**

- **Start tree**
  - **Construction method** Specify the tree construction method which should be used to create the initial tree, Neighbor Joining or 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 specify 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]
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 15.4.1.

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 15.9).

![Figure 15.9: Adjusting parameters for ML phylogeny](image)

To evaluate the reliability of the inferred trees, CLC Main Workbench allows the option of doing a bootstrap analysis (see section 15.4.1). 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.

### 15.4.1 Bioinformatics explained

#### The phylogenetic tree

The evolutionary hypothesis of a phylogeny can be graphically represented by a phylogenetic tree.

Figure 15.10 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.

![Figure 15.10: A proposed phylogeny of the great apes (Hominidae). Different components of the tree are marked, see text for description.](image)
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 15.10 is rooted at the most recent common ancestor of all Hominidae species, and therefore represents a hypothesis of the direction of evolution e.g. that the common ancestor of gorilla, chimpanzee and man existed before the common ancestor of chimpanzee and man. In contrast, an unrooted tree would represent relationships without assumptions about ancestry.

Besides evolutionary biology and systematics the inference of phylogenies is central to other areas of research.

As more and more genetic diversity is being revealed through the completion of multiple genomes, an active area of research within bioinformatics is the development of comparative machine learning algorithms that can simultaneously process data from multiple species [Siepel and Haussler, 2004]. Through the comparative approach, valuable evolutionary information can be obtained about which amino acid substitutions are functionally tolerant to the organism and which are not. This information can be used to identify substitutions that affect protein function and stability, and is of major importance to the study of proteins [Knudsen and Miyamoto, 2001]. Knowledge of the underlying phylogeny is, however, paramount to comparative methods of inference as the phylogeny describes the underlying correlation from shared history that exists between data from different species.

In molecular epidemiology of infectious diseases, phylogenetic inference is also an important tool. The very fast substitution rate of microorganisms, especially the RNA viruses, means that these show substantial genetic divergence over the time-scale of months and years. Therefore, the phylogenetic relationship between the pathogens from individuals in an epidemic can be resolved and contribute valuable epidemiological information about transmission chains and epidemiologically significant events [Leitner and Albert, 1999], [Forsberg et al., 2001].

Substitution models and distance estimation

When estimating the evolutionary distance between organisms, one needs a model of how frequently different mutations occur in the DNA. Such models are known as substitution models. Our Model Testing and Maximum Likelihood Phylogeny tools currently support the five nucleotide substitution models listed here:

- Jukes-Cantor [Jukes and Cantor, 1969]
- Felsenstein 81 [Felsenstein, 1981]
- Kimura 80 [Kimura, 1980]
- HKY [Hasegawa et al., 1985]
- GTR (also known as the REV model) [Yang, 1994a]

Common to all these models is that they assume mutations at different sites in the genome occur independently and that the mutations at each site follow the same common probability
distribution. Thus all five models provide relative frequencies for each of the 16 possible DNA substitutions (e.g. \(C \rightarrow A\), \(C \rightarrow C\), \(C \rightarrow G\),...).

The Jukes-Cantor and Kimura 80 models assume equal base frequencies and the HKY and GTR models allow the frequencies of the four bases to differ (they will be estimated by the observed frequencies of the bases in the alignment). In the Jukes-Cantor model all substitutions are assumed to occur at equal rates, in the Kimura 80 and HKY models transition and transversion rates are allowed to differ (substitution between two purines (\(A \leftrightarrow G\)) or two pyrimidines (\(C \leftrightarrow T\)) are transitions and purine - pyrimidine substitutions are transversions). The GTR model is the general time reversible model that allows all substitutions to occur at different rates. For the substitution rate matrices describing the substitution models we use the parametrization of Yang [Yang, 1994a].

For protein sequences, our Maximum Likelihood Phylogeny tool supports four substitution models:

- Bishop-Friday [Bishop and Friday, 1985]
- Dayhoff (PAM) [Dayhoff et al., 1978]
- JTT [Jones et al., 1992]
- WAG [Whelan and Goldman, 2001]

As with nucleotide substitution models, it is assumed that mutations at different sites in the genome occur independently and according to the same probability distribution.

The Bishop-Friday model assumes all amino acids occur with same frequency and that all substitutions are equally likely. This is the simplest model, but also the most unrealistic. The remaining three models use amino acid frequencies and substitution rates which have been determined from large scale experiments where huge sets of protein sequences have been aligned and rates have been estimated. These three models reflect the outcome of three different experiments. We recommend using WAG as these rates where estimated from the largest experiment.

**K-mer based distance estimation**

K-mer based distance estimation is an alternative to estimating evolutionary distance based on multiple alignments. At a high level, the distance between two sequences is defined by first collecting the set of k-mers (subsequences of length k) occuring in the two sequences. From these two sets, the evolutionary distance between the two organisms is now defined by measuring how different the two sets are. The more the two sets look alike, the smaller is the evolutionary distance. The main motivation for estimating evolutionary distance based on k-mers, is that it is computationally much faster than first constructing a multiple alignment. Experiments show that phylogenetic tree reconstruction using k-mer based distances can produce results comparable to the slower multiple alignment based methods [Blaisdell, 1989].

All of the k-mer based distance measures completely ignores the ordering of the k-mers inside the input sequences. Hence, if the selected k value (the length of the sequences) is too small, very distantly related organisms may be assigned a small evolutionary distance (in the extreme case where k is 1, two organisms will be treated as being identical if the frequency of each nucleotide/amino-acid is the same in the two corresponding sequences). In the other extreme,
the k-mers should have a length (k) that is somewhat below the average distance between mismatches if the input sequences were aligned (in the extreme case of k=the length of the sequences, two organisms have a maximum distance if they are not identical). Thus the selected k value should not be too large and not too small. A general rule of thumb is to only use k-mer based distance estimation for organisms that are not too distantly related.

**Formal definition of distance.** In the following, we give a more formal definition of the three supported distance measures: Euclidian-squared, Mahalanobis and Fractional common k-mer count. For all three, we first associate a point \( p(s) \) to every input sequence \( s \). Each point \( p(s) \) has one coordinate for every possible length k sequence (e.g. if \( s \) represent nucleotide sequences, then \( p(s) \) has \( 4^k \) coordinates). The coordinate corresponding to a length k sequence \( x \) has the value: "number of times \( x \) occurs as a subsequence in \( s \)". Now for two sequences \( s_1 \) and \( s_2 \), their evolutionary distance is defined as follows:

- **Euclidian squared:** For this measure, the distance is simply defined as the (squared Euclidian) distance between the two points \( p(s_1) \) and \( p(s_2) \), i.e.

  \[
  \text{dist}(s_1, s_2) = \sum_i (p(s_1)_i - p(s_2)_i)^2.
  \]

- **Mahalanobis:** This measure is essentially a fine-tuned version of the Euclidian squared distance measure. Here all the counts \( p(s_j)_i \), are "normalized" by dividing with the standard deviation \( \sigma_j \) 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 15.4.1).

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 reconstruct
trees by iteratively joining the two nearest clusters until there is only one cluster left. The
result of the UPGMA method is a rooted bifurcating tree annotated with branch lengths.

- The Neighbor Joining method [Saitou and Nei, 1987] attempts to reconstruct a minimum
evolution tree (a tree where the sum of all branch lengths is minimized). Opposite to the
UPGMA method, the 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 15.3) can be
used to test a range of different models for input nucleotide 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.

**15.5 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 15.6).

The preferred tree layout settings (user defined tree settings) can be saved and applied via the top right Save Tree Settings (figure 15.11). 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 15.11: Save, remove or apply preferred layout settings.

15.5.1 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 15.12). To navigate the tree using the Minimap, click on the Minimap with the mouse and move the grey square around within the Minimap.

Figure 15.12: Visualization of a phylogenetic tree. The grey square in the Minimap shows the part of the tree that is shown in the View Area.

15.5.2 Tree layout

The Tree Layout can be adjusted in the Side Panel (figure 15.13).
Figure 15.13: 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 15.13) or Decreasing.

- **Reset Tree Topology** Resets to the default tree topology and node order (see figure 15.13). Any previously collapsed nodes will be uncollapsed.
• **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.

### 15.5.3 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 15.14.

![Figure 15.14: The Node Layout settings. Node color is specified by metadata and is therefore inactive in this example.](image)

### 15.5.4 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 15.15).

• **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 15.5.9).

• **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 15.15: "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 15.15, only some of the labels are shown. The hidden labels are illustrated with thin horizontal lines (figure 15.16).

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 15.16 and figure 15.17). 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 15.5.5).

**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 15.16: The zoom function in the upper right corner of the 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 15.17: After zooming in on a region of interest more labels become visible. In this example all labels are now visible.

15.5.5 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.
15.5.6 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 15.18.

![Image of branch layout settings](image)

**Figure 15.18: Branch Layout settings.**

15.5.7 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 ≥ (%)** Highlights branches where the bootstrap value is above the user defined threshold.

### 15.5.8 Visualizing metadata

Metadata associated with a phylogenetic tree (described in detail in section 15.6) 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 15.19.
- **Label text color** The label text can be colored and used to visualize metadata (see figure 15.19).
- **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 15.14).

### 15.5.9 Node right click menu

Additional options for layout and extraction of subtree data are available when right clicking the nodes (figure 15.15):

- **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.
- **Collapse** Branches associated with a selected node can be collapsed with or without the associated labels. Collapsed branches can be uncollapsed using the **Uncollapse** option in the same menu.
Figure 15.19: 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.

- **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 15.20). When hiding nodes, a new button appears labeled "Show X hidden nodes" in the Side Panel under "Tree Layout" (figure 15.21). 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 15.11 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 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.
Figure 15.20: A subtree can be hidden by selecting "Hide Subtree" and is shown again when selecting "Show Hidden Subtree" on a parent node.

Figure 15.21: 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.

- **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

### 15.6 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 15.22).

![Figure 15.22: 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".](image)

Additional metadata can be associated with a tree by clicking the **Import Metadata** button. This will open up the dialog shown in figure 15.23.

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 15.23. 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 15.23: 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.]

**15.6.1 Table Settings and Filtering**

How to use the metadata table (see figure 15.24):

- **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 section 3.2.1).

**15.6.2 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 15.25):

- **Assign Metadata** The right click option "Assign Metadata" can be used for four purposes.
CHAPTER 15. PHYLOGENETIC TREES

Figure 15.24: 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.

Figure 15.25: Right click options in the metadata table.

- 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 points, 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 15.26). This will change values in all selected rows in the column that was clicked.

15.6.3 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
CHAPTER 15. PHYLOGENETIC TREES

Figure 15.26: To modify existing metadata, click on the specific field, select "Edit <column header>" and provide a new value.

Figure 15.27: To remove this entry in the legend, all nodes must have a value assigned in the corresponding metadata category.

Figure 15.27: 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.

15.6.4 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 15.28).

- **Selecting all nodes in a subtree** Double clicking on an 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 14.5.2). Next, change relevant option in the multiple alignment wizard that pops up and click **Finish**. The multiple alignment will now be generated.

![Figure 15.28: 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.](image-url)
Chapter 16

General sequence analyses

Contents

16.1 Annotate with GFF/GTF/GVF file ........................................... 339
16.2 Extract sequences ................................................................. 342
16.3 Shuffle sequence ................................................................. 343
16.4 Dot plots ............................................................................. 344
   16.4.1 Create dot plots .......................................................... 345
   16.4.2 View dot plots .............................................................. 345
   16.4.3 Bioinformatics explained: Dot plots .............................. 346
   16.4.4 Bioinformatics explained: Scoring matrices .................. 351
16.5 Local complexity plot ............................................................ 353
16.6 Sequence statistics ............................................................... 353
   16.6.1 Bioinformatics explained: Protein statistics ................. 356
16.7 Join Sequences ................................................................... 358
16.8 Pattern discovery ................................................................. 359
   16.8.1 Pattern discovery search parameters ......................... 360
   16.8.2 Pattern search output .................................................. 361
16.9 Motif Search ...................................................................... 361
   16.9.1 Dynamic motifs ........................................................... 361
   16.9.2 Motif search from the Toolbox .................................... 363
   16.9.3 Java regular expressions .............................................. 365
16.10 Create motif list ................................................................. 366

CLC Main Workbench offers different kinds of sequence analyses that apply to both protein and DNA.

The analyses are described in this chapter.

16.1 Annotate with GFF/GTF/GVF file

Use Annotate with GFF/GTF/GVF file to add annotations from a GFF3, GTF or GVF file onto a sequence, or sequences in a sequence list. The names in the first column in the file must match
the names of the sequences to be annotated. If this is not the case, either the names in the annotation file, or the names of the sequences, must be updated.

Tools are available for renaming sequences or sequences in sequence lists:

- Rename Elements, described in section 25.5
- Rename Sequences in Lists, described in section 25.6


### How annotations are applied

Annotations from each line in the annotation file are placed on the sequence with the name given in the first column. Special treatment is given to annotations of the types CDS, exon, mRNA, transcript and gene. For these, the following applies:

- A gene annotation is generated for each gene_id. The region annotated extends from the leftmost to the rightmost positions of all annotations that have the gene_id (gtf-style).

- CDS annotations that have the same transcriptID are joined to one CDS annotation (gtf-style). Similarly, CDS annotations that have the same parent are joined to one CDS annotation (gff-style).

- If there is more than one exon annotation with the same transcriptID these are joined to one mRNA annotation. If there is only one exon annotation with a particular transcriptID, and no CDS with this transcriptID, a transcript annotation is added instead of the exon annotation (gtf-style).

- Exon annotations that have the same mRNA as parent are joined to one mRNA annotation. Similarly, exon annotations that have the same transcript as parent, are joined to one transcript annotation (gff-style).

Note that genes and transcripts are linked by name only (not by position, ID etc).

### Running the tool

To run the Annotate with GFF/GTF/GVF file tool, go to:

Toolbox | General Sequence Analysis | Annotate with GFF/GTF/GVF file

After selecting the sequence to annotate, the next step will look like that shown in figure 16.1.

Click on Browse to select a GFF, GTF or GVF file. After working through handling options, described below, your sequences will be annotated by the information from that file.

### Name handling

Annotations are named in the following, prioritized way:

1. If one of the following qualifiers are present, it will be used for naming (prioritized):
CHAPTER 16. GENERAL SEQUENCE ANALYSES

Figure 16.1: Select a GFF, GTF or GVR file by clicking on the Browse button.

(a) Name
(b) Gene_name
(c) Gene_ID
(d) Locus_tag
(e) ID

2. If none of these are found, the annotation type will be used as name.

You can overrule this naming convention by choosing Replace all annotation names with this qualifier and specifying another qualifier (see figure 16.2).

If you provide a qualifier, it must be written identically to the corresponding qualifier name in the annotation file.

Transcript annotations are handled separately, since they inherit the name from the gene annotation.

Figure 16.2: You can choose Replace all annotation names with the specified qualifier.

Type handling
You can overrule feature types in the annotation file by choosing **Replace all annotation types with** and specifying a type to use.

**Ignore duplicate annotation**

When the **Ignore duplicate annotation** option is checked, only one instance of duplicate annotations will be added to the sequence.

**Create log**

In the Result handling section of the wizard, check the **Create log** box results to create a log that includes information like the number of annotations found and if there are any that are could not be placed on the sequence. This information can help with troubleshooting when annotations are not added to a sequence when they were expected to be.

### 16.2 Extract sequences

Extract Sequences extracts all the sequences from any of the element types below into a sequence list or into individual sequence elements:

- Alignments
- BLAST result
- BLAST overview tables
- Contigs and read mappings
- Read mapping tables
- Read mapping tracks
- RNA-Seq mapping results
- Sequence lists

If only a **subset** of the sequences are of interest, create an element containing just this subset first, and then run Extract Sequences on this. See the documentation for the relevant element types for further details. For example, for extracting a subset of a mapping, see section 19.7.6.

Paired reads are extracted in accordance with the read group settings, which are specified during the original import of the reads. If the orientation has since been changed (for example 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.

**Extracting sequences from sequence lists:** As all sequences will be extracted, the main reason to run this tool on a sequence list would be if you wished to create individual sequence elements from each sequence in the list. This is somewhat uncommon. If your aim is to create a list containing a subset of the sequences from another list, this can be done directly from the table view of sequence lists (see section 12.1.3), or using Split Sequence List (see section 25.4).
Running Extract Sequences

Launch Extract Sequences by going to:

Toolbox | General Sequence Analysis ( ) | Extract Sequences ( )

After selecting the elements to extract sequences from, you are offered the choice of extracting them to individual sequence elements or to a sequence list (figure 16.3). For most data types, a sequence list will be the best choice.

Below these options, the number of sequences that will be extracted is reported.

Figure 16.3: Extracted sequences can be put into a new sequence list or split into individual sequence elements.

16.3 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 16.4:

Figure 16.4: 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.

### 16.4 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.
16.4.1 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 16.5).

Figure 16.5: 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.

16.4.2 View dot plots

A view of a dot plot can be seen in figure 16.6. You can select Zoom in ( ) in the Toolbar and click the dot plot to zoom in to see the details of particular areas.

The Side Panel to the right let you specify the dot plot preferences. The gradient color box can be adjusted to get the appropriate result by dragging the small pointers at the top of the box. Moving the slider from the right to the left lowers the thresholds which can be directly seen in the dot plot, where more diagonal lines will emerge. You can also choose another color gradient by clicking on the gradient box and choose from the list.
Adjusting the sliders above the gradient box is also practical, when producing an output for printing where 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 (figure 16.7).

16.4.3 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.

CHAPTER 16. GENERAL SEQUENCE ANALYSES

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.

![Diagram of direct and inverted repeats](image)

Figure 16.9: Direct and inverted repeats shown on an amino acid sequence generated for demonstration purposes.

If the dot plot shows more than one diagonal in the same region of a sequence, the regions depending to the other sequence are repeated. In figure 16.10 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 16.11. In this figure, three frame shifts for the sequence on the y-axis are found.

1. Deletion of nucleotides
2. Insertion of nucleotides
3. Mutation (out of frame)
**Sequence inversions**  In dot plots you can see an inversion of sequence as contrary diagonal to the diagonal showing similarity. In figure 16.12 you can see a dot plot (window length is 3) with an inversion.

![Figure 16.12: The dot plot showing an inversion in a sequence. See also figure 16.9.](image)

**Low-complexity regions**  Low-complexity regions in sequences can be found as regions around the diagonal all obtaining a high score. Low complexity regions are calculated from the redundancy of amino acids within a limited region [Wootton and Federhen, 1993]. These are most often seen as short regions of only a few different amino acids. In the middle of figure 16.13 is a square shows the low-complexity region of this sequence.

![Figure 16.13: 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.](image)
16.4.4 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).

|   | A   | R   | N   | D   | C   | Q   | E   | G   | H   | I   | K   | M   | F   | P   | S   | T   | W   | Y   | V   |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| A | 4   | -1  | -2  | -2  | 0   | -1  | -1  | 0   | -2  | -1  | -1  | -1  | -1  | -1  | -1  | 0   | -3  | -2  | 0   |
| R | -1  | 5   | -2  | -3  | 1   | 0   | -2  | 0   | -3  | -2  | 1   | -3  | -2  | 1   | 0   | -4  | -2  | -3  |
| N | -2  | 0   | 6   | 1   | -3  | 0   | 0   | 0   | 1   | -3  | -3  | 0   | -2  | -3  | -2  | 1   | 0   | -2  | -3  |
| D | -2  | -2  | 1   | 6   | -3  | 0   | 2   | -1  | -1  | -3  | -4  | -1  | -3  | -3  | -1  | 0   | -1  | -4  | -3  |
| C | 0   | -3  | -3  | -3  | -3  | -3  | -4  | -3  | -3  | -1  | -1  | -2  | -3  | -1  | -1  | -1  | -2  | -1  | -2  |
| Q | -1  | 1   | 0   | 0   | -3  | 5   | 2   | -2  | 0   | -3  | 2   | 1   | 0   | -3  | 1   | 0   | -1  | -1  | 1   |
| E | -1  | 0   | 0   | 2   | -4  | 2   | 5   | -2  | 0   | -3  | -3  | 1   | 0   | -3  | 0   | 1   | -1  | -2  | 1   |
| G | 0   | -2  | 0   | -1  | -3  | -2  | -2  | 6   | -2  | 4   | -4  | -2  | 3   | 3   | 2   | 0   | -2  | -2  | -3  |
| H | -2  | 0   | 1   | -1  | -3  | 0   | 0   | -2  | 8   | -3  | -3  | -1  | -2  | -1  | -2  | 1   | -1  | -2  | -2  |
| I | 1   | -3  | -3  | -3  | -3  | -3  | -3  | -3  | -4  | -3  | 4   | 2   | 3   | 1   | 0   | -3  | 2   | -2  | -3  |
| L | -1  | -2  | -3  | -4  | -1  | -2  | -3  | -4  | -3  | 2   | 4   | -2  | 2   | 0   | 3   | -1  | -2  | -1  | -1  |
| K | 1   | -2  | 0   | -1  | 3   | 1   | 1   | 1   | -2  | -1  | -3  | 2   | 5   | -1  | 3   | 1   | 0   | -1  | -3  |
| M | -1  | -1  | 1   | -3  | -3  | -3  | -3  | -3  | -3  | -4  | 2   | 1   | 2   | -1  | 5   | 0   | -2  | -1  | -1  |
| F | -2  | -3  | -3  | -3  | -3  | -3  | -3  | -3  | -3  | 1   | 0   | 0   | 3   | 0   | 6   | 4   | -2  | -2  | 1   |
| P | -1  | -2  | -2  | -1  | -3  | -1  | -1  | -2  | 1   | -1  | -3  | -1  | -2  | 4   | 7   | -1  | -1  | -2  | -1  |
| S | 1   | 1   | 0   | -1  | 0   | 0   | 0   | -1  | -2  | -2  | 0   | -2  | 1   | -1  | 4   | 1   | -3  | -2  | -2  |
| T | 0   | -1  | 0   | -1  | -1  | -1  | -1  | -2  | -2  | -1  | -1  | -1  | 1   | -1  | 5   | 1   | -2  | 2   | 0   |
| W | -3  | -3  | -3  | -4  | -4  | -2  | -2  | -3  | -2  | -3  | -3  | -1  | -1  | 4   | 1   | -3  | 2   | 1   | 1   |
| Y | -2  | -2  | -2  | -3  | -2  | -1  | -2  | -3  | 2   | -1  | -2  | 1   | -1  | 3   | -3  | 2   | -2  | 2   | 7   |
| V | 0   | -3  | -3  | -3  | -1  | -2  | -3  | -3  | 3   | 1   | 2   | -1  | 1   | -2  | 2   | 0   | -3  | -1  | 4   |

Table 16.1: The BLOSUM62 matrix. A tabular view of the BLOSUM62 matrix containing all possible substitution scores [Henikoff and Henikoff, 1992].

Based on evolution of proteins it became apparent that these changes or substitutions of amino acids can be modeled by a scoring matrix also refereed to as a substitution matrix. See an example of a scoring matrix in table 16.1. This matrix lists the substitution scores of every single amino acid. A score for an aligned amino acid pair is found at the intersection of the corresponding column and row. For example, the substitution score from an arginine (R) to a lysine (K) is 2. The diagonal show scores for amino acids which have not changed. Most substitutions changes have a negative score. Only rounded numbers are found in this matrix.

The two most used matrices are the BLOSUM [Henikoff and Henikoff, 1992] and PAM [Dayhoff and Schwartz, 1978].

Different scoring matrices

- **PAM**

The first PAM matrix (Point Accepted Mutation) was published in 1978 by Dayhoff et al. The PAM matrix was 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 16.14).
There are some limitations 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 sequences where none of the aligned sequences share less than 62% identity. This resulted in a scoring matrix called BLOSUM62. In contrast to the PAM matrices the BLOSUM matrices are calculated from alignments without gaps emerging from the BLOCKS database [http://blocks.fhcrc.org/](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 16.14) for correlations between the PAM and BLOSUM matrices. To summarize, if you want to find distant related proteins to a sequence of interest using BLAST, you could benefit of using BLOSUM45 or similar matrices.

![Figure 16.14: 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.](http://blocks.fhcrc.org/)
16.5 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 16.15 shows an example of a local complexity plot.

Figure 16.15: 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.

16.6 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:
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 16.16), 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 16.16: Setting parameters for the Sequence statistics tool.](image)

For protein sequences, you can 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.](www.uniprot.org))

You can also choose between two different sets of values for calculation of extinction coefficients:

- **[Gill and von Hippel, 1989]:** Ext(Cystine) = 120, Ext(Tyr) = 1280 and Ext(Trp) = 5690
- **[Pace et al., 1995]:** Ext(Cystine) = 125, Ext(Tyr) = 1490 and Ext(Trp) = 5500

Read more about calculation of extinction coefficients in section section 16.6.1.

Click **Finish** to start the tool. An example of protein sequence statistics is shown in figure 16.17.

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:
  \[ \sum \text{units in sequence} \left( \frac{\text{weight}}{\text{unit}} \right) - \text{links} \times \frac{\text{weight}}{\text{H}_2\text{O}} \]
  where \( \text{links} \) is the sequence length minus one and \( \text{units} \) 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. This represents statistics for all codons; however, only codons that contribute with amino acids to the translated sequence will be counted.

A short description of the different areas of the statistical output is given in section 16.6.1.
16.6.1 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 post-translational 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:

  \[
  \text{Aliphatic index} = X(\text{Ala}) \times a + X(\text{Val}) \times b + X(\text{Leu}) \times b + X(Ile)
  \]

  \(X(\text{Ala}), X(\text{Val}), X(Ile)\) 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, simply determines the half-life of proteins. The estimated half-life of proteins have been investigated in mammals, yeast and E. coli (see Table 16.2). 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.

  Two values are reported. The first value, "Non-reduced cysteines", is computed assuming that all cysteine residues appear as half cystines, meaning they form di-sulfide bridges to other cysteines:
### Table 16.2: 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*.

<table>
<thead>
<tr>
<th>Amino acid</th>
<th>Mammalian</th>
<th>Yeast</th>
<th>E. coli</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ala (A)</td>
<td>4.4 hour</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Cys (C)</td>
<td>1.2 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Asp (D)</td>
<td>1.1 hours</td>
<td>3 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Glu (E)</td>
<td>1 hour</td>
<td>30 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Phe (F)</td>
<td>1.1 hours</td>
<td>3 min</td>
<td>2 min</td>
</tr>
<tr>
<td>Gly (G)</td>
<td>30 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>His (H)</td>
<td>3.5 hours</td>
<td>10 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Ile (I)</td>
<td>20 hours</td>
<td>30 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Lys (K)</td>
<td>1.3 hours</td>
<td>3 min</td>
<td>2 min</td>
</tr>
<tr>
<td>Leu (L)</td>
<td>5.5 hours</td>
<td>3 min</td>
<td>2 min</td>
</tr>
<tr>
<td>Met (M)</td>
<td>30 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Asn (N)</td>
<td>1.4 hours</td>
<td>3 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Pro (P)</td>
<td>&gt;20 hours</td>
<td>&gt;20 hours</td>
<td>?</td>
</tr>
<tr>
<td>Gln (Q)</td>
<td>0.8 hour</td>
<td>10 min</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Arg (R)</td>
<td>1 hour</td>
<td>2 min</td>
<td>2 min</td>
</tr>
<tr>
<td>Ser (S)</td>
<td>1.9 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Thr (T)</td>
<td>7.2 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Val (V)</td>
<td>100 hours</td>
<td>&gt;20 hours</td>
<td>&gt;10 hours</td>
</tr>
<tr>
<td>Trp (W)</td>
<td>2.8 hours</td>
<td>3 min</td>
<td>2 min</td>
</tr>
<tr>
<td>Tyr (Y)</td>
<td>2.8 hours</td>
<td>10 min</td>
<td>2 min</td>
</tr>
</tbody>
</table>

The extinction coefficient values of the three important amino acids at different wavelengths are found in [Gill and von Hippel, 1989] or in [Pace et al., 1995]. At 280nm the extinction coefficients are

- [Gill and von Hippel, 1989]: Ext(Cystine) = 120, Ext(Tyr) = 1280 and Ext(Trp) = 5690
- [Pace et al., 1995]: Ext(Cystine) = 125, Ext(Tyr) = 1490 and Ext(Trp) = 5500

This equation is only valid under the following conditions:

- pH 6.5
- 6.0 M guanidium hydrochloride
- 0.02 M phosphate buffer

Knowing the extinction coefficient, the absorbance (optical density) can be calculated using the following formula: $\text{Absorbance(Protein)} = \frac{\text{Ext(Protein)}}{\text{Molecular weight}}$
• **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.

### 16.7 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 | Classical Sequence Analysis | General Sequence Analyses | Join Sequences

This opens the dialog shown in figure 16.18.

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 16.19.
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 16.20.

Figure 16.20: The result of joining sequences is a new sequence containing the annotations of the joined sequences (they each had a HBB annotation).

16.8 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 16.21).

In order to search unknown sequences with an already existing model:
Select to use an already existing model which is seen in figure 16.21. Models are represented with the following icon in the Navigation Area ( ).

16.8.1 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 16.21.

- **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 16.22). If you have selected several sequences, a corresponding number of views will be opened.
16.8.2 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.

16.9 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 21.1.1). 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.

16.9.1 Dynamic motifs

In the Side Panel of sequence views, there is a group called Motifs (see figure 16.23).

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 16.24.

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 16.25.
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 16.23 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 16.10)). 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 16.26.

![Figure 16.26: Managing the motifs to be shown.](image)

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**.

### 16.9.2 Motif search from the Toolbox

The dynamic motifs described in section 16.9.1 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 16.27).

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 16.9.3.
  - Prosite regular expression. For proteins, you can enter different protein patterns from the PROSITE database (protein patterns using regular expressions and describing specific amino acid sequences). The PROSITE database contains a great number of patterns and have been used to identify related proteins (see [http://www.expasy.org/cgi-bin/prosite-list.pl](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 16.10).
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 16.9.3. 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 16.28.

Create table. This will create an overview table of all the motifs found for all the input sequences.
16.9.3 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 ` restricts the search to the beginning of your sequence. For example, if you search through a sequence with the regular expression `AC, the algorithm will find a match if AC occurs in the beginning of the sequence.

The symbol $ restricts the search to the end of your sequence. For example, if you search through a sequence with the regular expression GT$, the algorithm will find a match if GT occurs in the end of the sequence.

Examples
The expression [ACG][`AC]G{2} matches all strings of length 4, where the first character is A,C or G and the second is any character except A,C and the third and fourth character is G. The expression G.[`A]$ matches all strings of length 3 in the end of your sequence, where the first character is C, the second any character and the third any character except A.

16.10 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 16.29.

![Figure 16.29: Entering a new motif in the list.](image)

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 16.9.2 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 16.9.2.
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 16.9).
Chapter 17

Nucleotide analyses

Contents

17.1 Convert DNA to RNA ........................................... 368
17.2 Convert RNA to DNA ........................................... 369
17.3 Reverse complements of sequences ............................ 369
17.4 Translation of DNA or RNA to protein ......................... 370
17.5 Find open reading frames .................................... 372
    17.5.1 Open reading frame parameters ........................ 373

CLC Main Workbench offers different kinds of sequence analyses, which only apply to DNA and RNA.

17.1 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 17.1:

Figure 17.1: 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.

### 17.2 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](image)

This opens the dialog displayed in figure 17.2:

![Figure 17.2: Translating RNA to DNA.](image)

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.

### 17.3 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 17.3:

![Figure 17.3: Creating a reverse complement sequence.](image)

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.

### 17.4 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 17.4:

![Figure 17.4: Choosing sequences for translation.](image)

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
CHAPTER 17. NUCLEOTIDE ANALYSES

sequence lists from the selected elements.

Clicking **Next** generates the dialog seen in figure 17.5:

![Figure 17.5: Choosing translation of CDSs using standard translation table.](image)

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 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 17.5).

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 17.6): 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.

![Figure 17.6: Choosing how to translate CDS or ORF annotations.](image)

The CDS and ORF annotations are colored yellow as default.

### 17.5 Find open reading frames

The CLC Main Workbench Find Open Reading Frames function can be used to find all open reading frames (ORF) in a sequence, or, by choosing particular start codons to use, it can be used as a rudimentary gene finder. ORFs identified will be shown as annotations on the sequence. You have the option of choosing a translation table, the start codons to use, minimum ORF length as well as a few other parameters.

In one analysis, Find Open Reading Frames can process a maximum of 100,000 sequences or 50 million base pairs. Sequences may be provided to the tool as individual sequences or as sequence lists.

To find open reading frames:

Toolbox | Nucleotide Analysis | Find Open Reading Frames

This opens the dialog displayed in figure 17.7:

![Figure 17.7: Create Reading Frame dialog.](image)

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.

17.5.1 Open reading frame parameters

This opens the dialog displayed in figure 17.8:

![Find Open Reading Frames dialog](image)

Figure 17.8: Parameters for the Reading Frame tool.

The adjustable parameters for the search are:

- **Start codon**:
  - **AUG**. Most commonly used start codon.
  - **Any**. Find all open reading frames of specified length. Any combination of three bases that is not a stop-codon is interpreted as a start codon, and translated according to the specified genetic code.
  - **All start codons in genetic code**.
  - **Other**. Here you can specify a number of start codons separated by commas.

- **Both strands**. Finds reading frames on both strands.

- **Open-ended Sequence**. Allows the ORF to start or end outside the sequence. If the sequence studied is a part of a larger sequence, it may be advantageous to allow the ORF to start or end outside the sequence.

- **Genetic code translation table**.
• **Include stop codon in result** The ORFs will be shown as annotations which can include the stop codon if this option is checked. The translation tables are occasionally updated from NCBI. The tables are not available in this printable version of the user manual. Instead, the tables are included in the **Help** menu in the **Menu Bar** (in the appendix).

• **Minimum Length**. Specifies the minimum length for the ORFs to be found. The length is specified as number of codons.

Using open reading frames for gene finding is a fairly simple approach which is likely to predict genes which are not real. Setting a relatively high minimum length of the ORFs will reduce the number of false positive predictions, but at the same time short genes may be missed (see figure 17.9).

![Diagram](image.png)

**Figure 17.9:** The first 12,000 positions of the *E. coli* sequence NC_000913 downloaded from GenBank. The blue (dark) annotations are the genes while the yellow (brighter) annotations are the ORFs with a length of at least 100 amino acids. On the positive strand around position 11,000, a gene starts before the ORF. This is due to the use of the standard genetic code rather than the bacterial code. This particular gene starts with CTG, which is a start codon in bacteria. Two short genes are entirely missing, while a handful of open reading frames do not correspond to any of the annotated genes.

Click **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.
Chapter 18

Protein analyses

CLC Main Workbench offers a number of analyses of proteins as described in this chapter.


18.1 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 18.1:

![Figure 18.1: Choosing protein sequences to calculate protein charge.](image)

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 18.2 shows the electrical charges for three proteins. In the **Side Panel** to the right, you can modify the layout of the graph.

![Figure 18.2: View of the protein charge.](image)
18.2 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 18.3.

![Create Antigenicity Plot](image)

Figure 18.3: 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 chose from a number of antigenicity scales. Click **Finish** to start the tool. The result can be seen in figure 18.4.

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 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 18.3.1).

18.3 Hydrophobicity

CLC Main Workbench can calculate the hydrophobicity of protein sequences in different ways, using different algorithms (see section 18.3.2). 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**

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 18.5.

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 chose from a number of hydrophobicity scales which are further explained in section 18.3.2 Click Finish to start the tool. The result can be seen in figure 18.6.

See section A in the appendix for information about the graph view.
Figure 18.5: Step two in the Hydrophobicity Plot allows you to choose hydrophobicity scale and the window size.

Figure 18.6: The result of the hydrophobicity plot calculation and the associated Side Panel.

18.3.1 Hydrophobicity graphs along sequence

Hydrophobicity graphs along sequence can be displayed easily by activating the calculations from the Side Panel for a sequence. Simply right-click or double click on a protein sequence in the Navigation Area, and choose

Show | Sequence | open Protein info in Side Panel

These actions result in the view displayed in figure 18.7.

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 18.3.2).

In the following we will focus on the different ways that the Workbench offers to display the hydrophobicity scores. We use Kyte-Doolittle to explain the display of the scores, but the different options are the same for all the scales. Initially there are three options for displaying the
hydrophobicity scores. You can choose one, two or all three options by selecting the boxes (figure 18.8).

**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 18.8. Notice that you can choose the height of the graphs underneath the sequence.

**18.3.2 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.
CHAPTER 18. PROTEIN ANALYSES

Figure 18.9: 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 18.9).

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].
Table 18.1: Hydrophobicity scales. This table shows seven different hydrophobicity scales which are generally used for prediction of e.g. transmembrane regions and antigenicity.

- **Welling scale.** Welling et al. used information on the relative occurrence of amino acids in antigenic regions to make a scale which is useful for prediction of antigenic regions. This method is better than the Hopp-Woods scale of hydrophobicity which is also used to identify antigenic regions.

- **Kolaskar-Tongaonkar.** A semi-empirical method for prediction of antigenic regions has been developed [Kolaskar and Tongaonkar, 1990]. This method also includes information of surface accessibility and flexibility and at the time of publication the method was able to predict antigenic determinants with an accuracy of 75%.

- **Surface Probability.** Display of surface probability based on the algorithm by [Emini et al., 1985]. This algorithm has been used to identify antigenic determinants on the surface of proteins.

- **Chain Flexibility.** 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

18.4 Download Pfam Database

To run Pfam Domain Search you must first download the Pfam database. The Pfam database can be downloaded using:

Toolbox | Protein Analysis | Download Pfam Database

This element is designed for use with Pfam Domain Search. If you open it directly from the Navigation Area, only the element history is accessible.

18.5 Pfam domain search

Pfam Domain Search searches for domains in protein sequences using the Pfam database [Bate-man et al., 2004], a large collection of multiple sequence alignments and hidden Markov models (HMMs) covering many common protein domains. It can add Region annotations on the input sequences where domains were found (figure 18.11) and it can output a table listing the domains found.

Why search against Pfam? Many proteins have a unique combination of domains, which can be responsible for 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.

After the Pfam database has been downloaded (see section 18.4), start Pfam Domain Search by going to:

Toolbox | Protein Analysis | Pfam Domain Search

By selecting several input sequences, you can perform the analysis on all these at once. Options can be configured (figure 18.10).

Pfam Domain Search options

- **Database** Choose the database to use when searching for Pfam domains.
- **Significance cutoff:**
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 than 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.

If annotations were added but are not initially visible on your sequences, check under the "Annotation types" tab of the side panel settings to ensure the Region annotation type has been checked.

![Image of annotations added by Pfam search tool]

Figure 18.11: Annotations (in red) that were added by the Pfam search tool.

Detailed information for each domain annotation is available in the annotation tool tip as well as in the Annotation Table view of the sequence list.

The domain search is performed using the hmmssearch tool from the HMMER3 package version 3.1b1 [http://hmmer.org/]. Detailed information about the scores in the Region annotations added can be found in the HMMER User Guide [http://eddylab.org/software/hmmer/Userguide.pdf].

Individual domain annotations can be removed manually, if desired. See section 12.3.5.

18.6 Download 3D Protein Structure Database

This tool downloads the 3D Protein Structure Database from a public accessible HTTP location hosted by QIAGEN Aarhus.

The database contains a curated set of sequences with known 3D structures, which are obtained from the Protein Data Bank ([http://www.wwpdb.org](http://www.wwpdb.org)) [Berman et al., 2003]. The information stored in the database (e.g. protein sequence, X-ray resolution) is used to identify suitable structural templates when using the Link Variants to 3D Protein Structure tool.

To download the database, go to:

**Toolbox | Protein Analysis | Download 3D Protein Structure Database**
If you are connected to a server, you will first be asked about whether you want to download the data locally or on a server. In the next wizard step you are asked to select the download location (see figure 18.12).

![Figure 18.12: Select the download location.](image)

The downloaded database will be installed in the same location as local BLAST databases (e.g. `<username>/CLCdatabases`) or at a server location if the tool was executed on a CLC Server. From the wizard it is possible to select alternative locations if more than one location is available.

When new databases are released, a new version of the database can be downloaded by invoking the tool again (the existing database will be replaced).

If needed, the Manage BLAST Databases tool can be used to inspect or delete the database (the database is listed with the name ‘ProteinStructureSequences’). You can find the tool here:

BLAST | Manage BLAST Databases

### 18.7 Find and Model Structure

This tool is used to find suitable protein structures for representing a given protein sequence. From the resulting table, a structure model (homology model) of the sequence can be created by one click, using one of the found protein structures as template.

To run the Find and Model Structure tool:

**Toolbox | Sequence Analysis | Find and Model Structure**

**Note:** Before running the tool, a protein structure sequence database must be downloaded and installed using the ‘Download Find Structure Database’ tool (see section 18.6).

In the tool wizard step 1, select the amino acid sequence to use as query from the Navigation Area.

In step 2, specify if the output table should be opened or saved.

The Find and Model Structure tool carries out the following steps, to find and rank available structures representing the query sequence:

**Input:** Query protein sequence

1. BLAST against protein structure sequence database
2. Filter away low quality hits
3. Rank the available structures

**Output:** Table listing available structures

In the output table (figure 18.13), the column named “Available Structures” contains links that will invoke a menu with the options to either create a structure model of the query sequence or just download the structure. This is further described in section 18.7.1. The remaining columns contain additional information originating from the PDB file or from the BLAST search.

![Figure 18.13: Table output from Find and Model Structure.](image)

The three steps carried out by the *Find and Model Structure* tool are described in short below.

**BLAST against protein structure sequence database** A local BLAST search is carried out for the query sequence against the protein structure sequence database (see section 18.6). BLAST hits with E-value > 0.0001 are rejected and a maximum of 2500 BLAST hits are retrieved. Read more about BLAST in section 24.5.

**Filter away low quality hits** From the list of BLAST hits, entries are rejected based on the following rules:

- PDB structures with a resolution lower than 4 Å are removed since they cannot be expected to represent a trustworthy atomistic model.
- BLAST hits with an identity to the query sequence lower than 20 % are removed since they most likely would result in inaccurate models.

**Rank the available structures** For the resulting list of available structures, each structure is scored based on its homology to the query sequence, and the quality of the structure itself. The *Template quality score* is used to rank the structures in the table, and the rank of each structure is shown in the "Rank" column (see figure 18.13). Read more about the *Template quality score* in section 18.7.2.

### 18.7.1 Create structure model

Clicking on a link in the "Available structures" column will show a menu with three options:

- Download and Open
- Download and Create Model
Help

The "Download and Open" option will do the following:

1. **Download and import** the PDB file containing the structure.
2. **Create an alignment** between the query and structure sequences.
3. **Open a 3D view** (Molecule Project) with the molecules from the PDB file and open the created sequence alignment. The sequence originating from the structure will be linked to the structure in the 3D view, so that selections on the sequence will show up on the structure (see section 13.4).

The "Download and Create Model" option will do the following:

1. **Download and import** the PDB file containing the structure.
2. **Generate a biomolecule** involving the protein chain to be modeled. Biomolecule information available in the template PDB file is used (see section 13.6). If several biomolecules involving the chain are available, the first one is applied.
3. **Create an alignment** between the query and structure sequences.
4. **Create a model structure** by mapping the query sequence onto the structure based on the sequence alignment (see section 18.7.2). If multiple copies of the template protein chain have been made to generate a biomolecule, all copies are modeled at the same time.
5. **Open a 3D view** (a Molecule Project) with the structure model shown in both backbone and wireframe representation. The model is colored by temperature (see figure 18.14), to indicate local model uncertainty (see section 18.7.2). Other molecules from the template PDB file are shown in orange or yellow coloring. The created sequence alignment is also opened and linked with the 3D views so that selections on the model sequence will show up on the model structure (see section 13.4).

The template structure is also available from the Proteins category in the Project Tree, but hidden in the initial view. The initial view settings are saved on the Molecule Project as "Initial visualization", and can always be reapplied from the View Settings menu found in the bottom right corner of the Molecule Project (see section 4.6).

If you have problems viewing 3D structures, please check your system matches the requirements for 3D Viewers. See section 1.3.

### 18.7.2 Model structure

**Protein coloring to visualize local structural uncertainties**

The default coloring scheme for modeled structures in CLC Main Workbench is "Color by Temperature". This coloring indicates the uncertainty or disorder of each atom position in the structure.
Figure 18.14: Structure Model of CDK5_HUMAN. The atoms and backbone are colored by temperature, showing uncertain structure in red and well defined structure in blue.

For crystal structures, the temperature factor (also called the B-factor) is given in the PDB file as a measure of the uncertainty or disorder of each atom position. The temperature factor has the unit $\text{Å}^2$, and is typically in the range $[0, 100]$.

The temperature color scale ranges from blue (0) over white (50) to red (100) (see section 13.3.1).

For structure models created in CLC Main Workbench, the temperature factor assigned to each atom combines three sources of positional uncertainty:

- **PDB Temp.** The atom position uncertainty for the template structure, represented by the temperature factor of the backbone atoms in the template structure.

- **P(alignment)** The probability that the alignment of a residue in the query sequence to a particular position on the structure is correct.

- **Clash?** It is evaluated if atoms in the structure model seem to clash, thereby indicating a problem with the model.

The three aspects are combined to give a temperature value between zero and 100, as illustrated in figure 18.15 and 18.16.

When holding the mouse over an atom, the Property Viewer in the Side Panel will show various information about the atom. For atoms in structure models, the contributions to the assigned temperature are listed as seen in figure 18.17.

**Note:** For NMR structures, the temperature factor is set to zero in the PDB file, and the "Color by Temperature" will therefore suggest that the structure is more well determined than is actually the case.

**P(alignment)** Alignment error is one of the largest causes of model inaccuracy, particularly when the model is built from a template sharing low sequence identity (e.g. lower than 60%).
Misaligning a single amino acid by one position will cause a ca. 3.5 Å shift of its atoms from their true positions.

The estimate of the probability that two amino acids are correctly aligned, \( P(\text{alignment}) \), is obtained by averaging over all the possible alignments between two sequences, similar to [Knudsen and Miyamoto, 2003].

This allows local alignment uncertainty to be detected even in similar sequences. For example the position of the D in this alignment:

\[
\text{Template} \quad \text{GGACDAEDRSTRSTACE---GG} \\
\text{Target} \quad \text{GGACD---RSTRSTACEKLMGG}
\]
is uncertain, because an alternative alignment is as likely:

<table>
<thead>
<tr>
<th>Template</th>
<th>GGACDAEDRSTRSTACE---GG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target</td>
<td>GGAC---DRSTRSTACEKLMGG</td>
</tr>
</tbody>
</table>

**Clash?** Clashes are evaluated separately for each atom in a side chain. If the atom is considered to clash, it will be assigned a temperature of 100.

**Note:** Clashes within the modeled protein chain as well as with all other molecules in the downloaded PDB file (except water) are considered.

**Ranking structures**

The protein sequence of the gene affected by the variant (the query sequence) is BLASTed against the protein structure sequence database (section 18.6).

A *template quality score* is calculated for the available structures found for the query sequence. The purpose of the score is to rank structures considering both their quality and their homology to the query sequence.

The five descriptors contributing to the score are:

- E-value
- % Match identity
- % Coverage
- Resolution (of crystal structure)
- Free R-value ($R_{free}$ of crystal structure)

Each of the five descriptors are scaled to [0,1], based on the linear functions seen in figure 18.18. The five scaled descriptors are combined into the *template quality score*, weighting them to emphasize homology over structure qualities.

$$\text{Template quality score} = 3 \cdot S_{E\text{-value}} + 3 \cdot S_{\text{Identity}} + 1.5 \cdot S_{\text{Coverage}} + S_{\text{Resolution}} + 0.5 \cdot S_{R_{\text{free}}}$$

**E-value** is a measure of the quality of the match returned from the BLAST search. You can read more about BLAST and E-values in section 24.5.

**% Match identity** is the identity between the query sequence and the BLAST hit in the matched region. It is evaluated as

$$\text{% Match identity} = 100\% \cdot \frac{\text{Identity in BLAST alignment}}{L_B}$$

where $L_B$ is the length of the BLAST alignment of the matched region, as indicated in figure 18.19, and “Identity in BLAST alignment” is the number of identical positions in the matched region.

**% Coverage** indicates how much of the query sequence has been covered by a given BLAST hit (see figure 18.19). It is evaluated as
From the E-value, % Match identity, % Coverage, Resolution, and Free R-value, the contributions to the "Template quality score" are determined from the linear functions shown in the graphs.

\[ \text{% Coverage} = 100\% \cdot \frac{(L_B - L_G)}{L_Q} \]

where \( L_G \) is the total length of gaps in the BLAST alignment and \( L_Q \) is the length of the query sequence.

Figure 18.19: Schematic of a query sequence matched to a BLAST hit. \( L_Q \) is the length of the query sequence, \( L_B \) is the length of the BLAST alignment of the matched region, QG1-3 are gaps in the matched region of the query sequence, HG1-2 are gaps in the matched region of the BLAST hit sequence, \( L_G \) is the total length of gaps in the BLAST alignment.

The resolution of a crystal structure is related to the size of structural features that can be resolved from the raw experimental data.

\( R_{\text{free}} \) is used to assess possible overmodeling of the experimental data.

Resolution and \( R_{\text{free}} \) are only given for crystal structures. NMR structures will therefore usually
be ranked lower than crystal structures. Likewise, structures where $R_{\text{free}}$ has not been given will tend to receive a lower rank. This often coincides with structures of older date.

**How a model structure is created**

A structure model is created by mapping the query sequence onto the template structure based on a sequence alignment (see figure 18.20):

![Figure 18.20: Sequence alignment mapping query sequence (Query CDK5_HUMAN) to the structure with sequence "Template(3OJQ - CYCLIN-DEPENDENT KINASE 2)", producing a structure with sequence "Model(CDK5_HUMAN)". Examples are highlighted: 1. Identical amino acids, 2. Amino acid changes, 3. Amino acids in query sequence not aligned to a position on the template structure, and 4. Amino acids on the template structure, not aligned to query sequence.]

- For identical amino acids (example 1 in figure 18.20) => Copy atom positions from the PDB file. If the side chain is missing atoms in the PDB file, the side chain is rebuilt (section 18.7.2).
- For amino acid changes (example 2 in figure 18.20) => Copy backbone atom positions from the PDB file. Model side chain atom positions to match the query sequence (section 18.7.2).
- For amino acids in the query sequence not aligned to a position on the template structure (example 3 in figure 18.20) => No atoms are modeled. The model backbone will have a gap at this position and a "Structure modeling" issue is raised (see section 13.1.4).
- For amino acids on the template structure, not aligned to the query sequence (example 4 in figure 18.20) => The residues are deleted from the structure and a "Structure modeling" issue is raised (see section 13.1.4).

**How side chains are modeled**

Amino acid side chains tend to assume one of a discrete number of "rotamer" conformations. The rotamers used in CLC Main Workbench have been calculated from a non-redundant set of high-resolution crystal structures.

Side chains are modeled using a heat bath Monte Carlo simulated annealing algorithm, similar to the OPUS-Rota method [Lu et al., 2008]. The algorithm consists of approximately 100 cycles of simulation. In a single cycle, rotamers are selected for each side chain with a probability
according to their energy. As the simulation proceeds, the selection increasingly favors the rotamers with the lowest energy, and the algorithm converges.

A local minimization of the modeled side chains is then carried out, to reduce unfavorable interactions with the surroundings.

**Calculating the energy of a side chain rotamer**

The total energy is composed of several terms:

- **Statistical potential**: This score accounts for interactions between the given side chain and the local backbone, and is estimated from a database of high-resolution crystal structures. It depends only on the rotamer and the local backbone dihedral angles $\phi$ and $\psi$.

- **Atom interaction potential**: This score is used to evaluate the interaction between a given side chain atom and its surroundings.

- **Disulfide potential**: Only applies to cysteines. It follows the form used in the RASP program [Miao et al., 2011] and serves to allow disulfide bridges between cysteine residues. It penalizes deviations from ideal disulfide geometry. A distance filter is applied to determine if the disulfide potential should be used, and when it is applied the atom interaction potential between the two sulfur atoms is turned off. Note that disulfide bridges are not formed between separate chains.

**Note**: The atom interaction potential considers interactions within the modeled protein chain as well as with all other molecules in the downloaded PDB file (except water).

**Local minimization of side chain**

After applying a side chain rotamer from the library to the backbone, a local minimization may be carried out for rotations around single bonds in the side chain.

The potential to minimize with respect to bond rotation is composed of the following terms:

- **Atom interaction potential**: Same as for calculating the energy of a rotamer.

- **Disulfide potential**: Same as for calculating the energy of a rotamer.

- **Harmonic potential**: This penalizes small deviations from ideal rotamers according to a harmonic potential. This is motivated by the concept of a rotamer representing a minimum energy state for a residue without external interactions.

**18.8 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 shows 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 18.21:

![Image](image1.png)

**Figure 18.21: 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 18.22).

![Image](image2.png)

**Figure 18.22: Alpha-helices and beta-strands shown as annotations on the sequence.**

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 12.3.2.

Undesired alpha-helices or beta-sheets can be removed through the **Delete Annotation ( )** right-click mouse menu. See section 12.3.5.
18.9 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 16.6.1.
- **Protein charge plot.** Plot of charge as function of pH, see section 18.1.
- **Hydrophobicity plot.** See section 18.3.
- **Complexity plot.** See section 16.5.
- **Dot plot.** See section 16.4.
- **Secondary structure prediction.** See section 18.8.
- **Pfam domain search.** See section 18.5.
- **BLAST against NCBI databases.** See section 24.1.1.

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](http://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 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 24.1.1.

An example of Protein report can be seen in figure 18.23.

![Figure 18.23: A protein report. There is a Table of Contents in the Side Panel that makes it easy to browse the report.](image)

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.
18.10 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 18.10.1.

In order to make a reverse translation:

Toolbox | Protein Analysis | Reverse Translate

This opens the dialog displayed in figure 18.24:

![Figure 18.24: Choosing a protein sequence for reverse translation.](image)

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 18.25:

![Figure 18.25: Choosing parameters for the reverse translation.](image)

- **Use random codon.** This will randomly back-translate an amino acid to a codon assuming 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/](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.

### 18.10.1 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/](http://nobelprize.org/medicine/laureates/1968/)). The Genetic Code represents translations of all 64 different codons into 20 different amino acids. Therefore it is no problem to translate a DNA/RNA sequence into a specific protein. But due to the degeneracy of the genetic code, several codons may code for only one specific amino acid. This can be seen in the table below. After the discovery of the genetic code it has been concluded that different organism (and organelles) have genetic codes which are different from the "standard genetic code". Moreover, the amino acid alphabet is no longer limited to 20 amino acids. The 21’st amino acid, selenocysteine, is encoded by an 'UGA' codon which is normally a stop codon. The
discrimination of a selenocysteine over a stop codon is carried out by the translation machinery. Selenocysteines are very rare amino acids.

The table below shows the Standard Genetic Code which is the default translation table.

<table>
<thead>
<tr>
<th>Codon</th>
<th>Amino Acid</th>
<th>Codon</th>
<th>Amino Acid</th>
<th>Codon</th>
<th>Amino Acid</th>
</tr>
</thead>
<tbody>
<tr>
<td>TTT F</td>
<td>Phe</td>
<td>TCT S</td>
<td>Ser</td>
<td>TAT Y</td>
<td>Tyr</td>
</tr>
<tr>
<td>TTC F</td>
<td>Phe</td>
<td>TCC S</td>
<td>Ser</td>
<td>TAC Y</td>
<td>Tyr</td>
</tr>
<tr>
<td>TTA L</td>
<td>Leu</td>
<td>TCA S</td>
<td>Ser</td>
<td>TAA *</td>
<td>Ter</td>
</tr>
<tr>
<td>TTG L</td>
<td>Leu</td>
<td>TCG S</td>
<td>Ser</td>
<td>TAG *</td>
<td>Ter</td>
</tr>
<tr>
<td>CTT L</td>
<td>Leu</td>
<td>CCT P</td>
<td>Pro</td>
<td>CAT H</td>
<td>His</td>
</tr>
<tr>
<td>CTC L</td>
<td>Leu</td>
<td>CCC P</td>
<td>Pro</td>
<td>CAC H</td>
<td>His</td>
</tr>
<tr>
<td>CTA L</td>
<td>Leu</td>
<td>CCA P</td>
<td>Pro</td>
<td>CAA Q</td>
<td>Gln</td>
</tr>
<tr>
<td>CTG L</td>
<td>Leu</td>
<td>CCG P</td>
<td>Pro</td>
<td>CAG Q</td>
<td>Gln</td>
</tr>
<tr>
<td>ATT I</td>
<td>Ile</td>
<td>ACT T</td>
<td>Thr</td>
<td>AAT N</td>
<td>Asn</td>
</tr>
<tr>
<td>ATC I</td>
<td>Ile</td>
<td>ACC T</td>
<td>Thr</td>
<td>AAC N</td>
<td>Asn</td>
</tr>
<tr>
<td>ATA I</td>
<td>Ile</td>
<td>ACA T</td>
<td>Thr</td>
<td>AAA K</td>
<td>Lys</td>
</tr>
<tr>
<td>ATG M</td>
<td>Met</td>
<td>ACG T</td>
<td>Thr</td>
<td>AAG K</td>
<td>Lys</td>
</tr>
<tr>
<td>GTT V</td>
<td>Val</td>
<td>GCT A</td>
<td>Ala</td>
<td>GAT D</td>
<td>Asp</td>
</tr>
<tr>
<td>GTC V</td>
<td>Val</td>
<td>GCC A</td>
<td>Ala</td>
<td>GAC D</td>
<td>Asp</td>
</tr>
<tr>
<td>GTA V</td>
<td>Val</td>
<td>GCA A</td>
<td>Ala</td>
<td>GAA E</td>
<td>Glu</td>
</tr>
<tr>
<td>GTG V</td>
<td>Val</td>
<td>GCG A</td>
<td>Ala</td>
<td>GAG E</td>
<td>Glu</td>
</tr>
</tbody>
</table>

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:
Codon usage database:
http://www.kazusa.or.jp/codon/
Wikipedia on the genetic code
http://en.wikipedia.org/wiki/Genetic_code

18.11 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 18.26. You can select one or several sequences.

![Figure 18.26: Choosing a protein sequence for proteolytic cleavage.](image)

In the second dialog, you can select proteolytic cleavage enzymes. Presently, the list contains the enzymes shown in figure 18.27. The full list of enzymes and their cleavage patterns can be seen in Appendix, section C.

You can then set parameters for the detection. This limits the number of detected cleavages (figure 18.28).

- **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 18.29.

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.
18.11.1 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 18.30.

Figure 18.30: 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 18.31).

Figure 18.31: 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.

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/
Chapter 19

Sequencing data analyses and Assembly

Contents

19.1 Importing and viewing trace data ............................................. 405
  19.1.1 Trace settings in the Side Panel ......................................... 405
19.2 Trim sequences ................................................................. 406
  19.2.1 Trimming using the Trim tool ........................................... 407
  19.2.2 Manual trimming ............................................................ 409
19.3 Assemble sequences ............................................................. 409
19.4 Assemble sequences to reference .............................................. 411
19.5 Sort sequences by name ......................................................... 414
19.6 Add sequences to an existing contig .......................................... 417
19.7 View and edit contigs and read mappings .................................... 418
  19.7.1 View settings in the Side Panel .......................................... 418
  19.7.2 Editing a contig or read mapping ....................................... 423
  19.7.3 Sorting reads ............................................................... 423
  19.7.4 Read conflicts ............................................................. 424
  19.7.5 Using the mapping ......................................................... 424
  19.7.6 Extract reads from a mapping .......................................... 424
  19.7.7 Variance table ............................................................. 427
19.8 Reassemble contig ............................................................... 428
19.9 Secondary peak calling ......................................................... 429
19.10 Extract Consensus Sequence .................................................. 430
19.11 Combine Reports ............................................................... 433
  19.11.1 Combine Reports output ................................................. 434

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 7.1).

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.


19.1 Importing and viewing trace data

A number of different binary trace data formats can be imported into the program, including Standard Chromatogram Format (.SCF), ABI sequencer data files ( .ABI and .AB1), PHRED output files (.PHD) and PHRAP output files (.ACE) (see section 7.1).

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 19.1.

![Figure 19.1: A tooltip displaying information about the quality of the chromatogram.](image)

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.

The traces can be scaled by dragging the trace vertically as shown in figure 19.2. 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 19.1.1.

![Figure 19.2: Grab the traces to scale.](image)

19.1.1 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 19.3):

- **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 19.1.
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.

### 19.2 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 19.4). 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 19.3). 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 12.3.2).

Figure 19.4: *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.

### 19.2.1 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 Sequences tool in the Workbench, go to the menu option:

**Toolbox | Sanger Sequencing Analysis | Trim Sequences**

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 19.5.

![Trim Sequences Dialog](image)

**Figure 19.5: 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: 
\[
p_{\text{error}} = 10^{-0.1Q}.
\]
(This now means that low values are high quality bases.)

Next, for every base a new value is calculated: \( \text{Limit} - p_{\text{error}} \). 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.

At [http://resources.qiagenbioinformatics.com/testdata/trim.zip](http://resources.qiagenbioinformatics.com/testdata/trim.zip) you find an example sequence and an Excel sheet showing the calculations done for this particular sequence to illustrate the procedure described above.

• **Trim ambiguous nucleotides.** This option trims the sequence ends based on the presence of ambiguous nucleotides (typically N). Note that the automated sequencer generating the data must be set to output ambiguous nucleotides in order for this option to apply. The algorithm takes as input the *maximal number of ambiguous nucleotides allowed in the sequence after trimming*. If this maximum is set to e.g. 3, the algorithm finds the maximum length region containing 3 or fewer ambiguities and then trims away the ends not included in this region. 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 UniVec database build 10.0 is included when you install the CLC Main Workbench. A list of all the vectors in the database can be found at [http://www.ncbi.nlm.nih.gov/VeScree/replist.html](http://www.ncbi.nlm.nih.gov/VeScree/replist.html).

• **Trim contamination from sequences.** This option lets you use your own vector sequences that you have imported into the CLC Main Workbench. If selected, **Trim using sequences** will be enabled and you can choose one or more sequences.

• **Hit limit for vector trimming.** When at least one vector trimming parameter is selected, the strictness for vector contamination trimming can be specified. 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/VeScree/VeScree.html](http://www.ncbi.nlm.nih.gov/VeScree/VeScree.html)):
– **Weak hit limit** 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 hit limit** 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 hit limit** Expect 1 random match in 1,000,000 queries of length 350 kb.
  * Terminal match with Score at least 24.
  * Internal match with Score at least 30.

In the last step of the wizard, you can choose to create a report, summarizing how each sequence has been trimmed. 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.

### 19.2.2 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 Sequences tool) as regions to be ignored.

### 19.3 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 19.4).

**Note!** You can assemble a maximum of 10,000 sequences at a time.

Tools to assemble a larger number of sequences are available in the **CLC Genomics Workbench**: https://digitalinsights.qiagen.com/products-overview/discovery-insights-portfolio/analysis-and-visualization/qiagen-clc-genomics-workbench/

To perform the assembly:

**Toolbox | Sanger Sequencing Analysis ( ) | Assemble Sequences ( )**

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 19.6

![Figure 19.6: Setting assembly parameters.](image)

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 19.7.4.

- **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 19.7.

- **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 19.7.7.

- **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 19.7 on how to use the resulting contigs.

## 19.4 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.

Tools to assemble a larger number of sequences are available in the CLC Genomics Workbench: https://digitalinsights.qiagen.com/products-overview/discovery-insights-portfolio/analysis-and-visualization/qiagen-clc-genomics-workbench/

To start the assembly:


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 19.7

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 19.8.

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):

<table>
<thead>
<tr>
<th>Score values</th>
<th>Low</th>
<th>Medium</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Match (mt)</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Transversion (tv)</td>
<td>-6</td>
<td>-10</td>
<td>-20</td>
</tr>
<tr>
<td>Transition (ti)</td>
<td>-2</td>
<td>-6</td>
<td>-16</td>
</tr>
<tr>
<td>Unknown (un)</td>
<td>-2</td>
<td>-6</td>
<td>-16</td>
</tr>
<tr>
<td>Gap</td>
<td>-8</td>
<td>-16</td>
<td>-36</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Score Matrix</th>
<th>A</th>
<th>C</th>
<th>G</th>
<th>T</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>mt</td>
<td>tv</td>
<td>ti</td>
<td>tv</td>
<td>un</td>
</tr>
<tr>
<td>C</td>
<td>tv</td>
<td>mt</td>
<td>tv</td>
<td>ti</td>
<td>un</td>
</tr>
<tr>
<td>G</td>
<td>ti</td>
<td>tv</td>
<td>mt</td>
<td>tv</td>
<td>un</td>
</tr>
<tr>
<td>T</td>
<td>tv</td>
<td>ti</td>
<td>tv</td>
<td>mt</td>
<td>un</td>
</tr>
<tr>
<td>N</td>
<td>un</td>
<td>un</td>
<td>un</td>
<td>un</td>
<td>un</td>
</tr>
</tbody>
</table>

• **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 19.2 for more information about trimming).

Click **Finish** to start the tool. This will start the assembly process. See section 19.7 on how to use the resulting contigs.

### 19.5 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:

```plaintext
...  
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:

```plaintext
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 19.9.

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 19.9.
Advanced splitting using regular expressions

You can see a more detail explanation of the regular expressions syntax in section 16.9.3.

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 19.9 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 19.10.
CHAPTER 19. SEQUENCING DATA ANALYSES AND ASSEMBLY

Figure 19.10: 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.

19.6 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 ( ) | Add Sequences to Contig ( )

or right-click in the empty white area of the contig | Add Sequences to Contig ( )

This opens a dialog where you can 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 19.2.1).

When the elements are selected, click Next, and you will see the dialog shown in figure 19.11.
CHAPTER 19. SEQUENCING DATA ANALYSES AND ASSEMBLY

Figure 19.11: 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 19.4).

Click Finish to start the tool. This will start the assembly process. See section 19.7 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.

19.7 View and edit contigs and read mappings

The results of Assemble Sequences or Assemble Sequences to Reference are contigs or read mappings, respectively (figure 19.12). A mapping is generated for each reference supplied as input to Assemble Sequences to Reference. Individual read mappings can be opened by double-clicking on a row in the table that lists the mappings.

Customizing how a contig or mapping looks is covered in section 19.7.1.

The residue colors and traces at the ends of reads are faded in regions that do not contribute to the final contig or mapping results. This may be due to trimming before or during the assembly or to misalignment with other reads (assembly) or the reference sequence (mapping).

Simply drag the edge of the faded section to adjust the trimmed area to include more of the read in the contig or mapping (figure 19.13).

Note: Handles for dragging are only available when individual residues can be seen. For this, zoom fully in and chose a Compactness level of "Not compact", "Low" or "Packed".

To reverse complement an entire contig or mapping, right-click in the empty white area of the contig or mapping and choose to Reverse Complement Sequence.

19.7.1 View settings in the Side Panel

Here we cover a few of the key settings available in the side panel for customizing how a contig or read mapping looks. We step through them according to the tab names in the side panel.

Read layout.
Figure 19.12: The view of a contig. Controls at the bottom allow you to zoom in and out, and settings to the right control how the mapping is displayed.

Figure 19.13: Drag the edge of the faded area to customize how much of a read should be considered in the mapping.

- **Compactness.** Set the level of detail to be displayed. The level of compactness affects other view settings as well as the overall view. For example: if Compact is selected, quality scores and annotations on the reads will not be visible, even if these options are turned on under the "Nucleotide info" palette. Compactness can also be changed by pressing and holding the Alt key while scrolling 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 present. To view such information, additional viewing options under the Nucleotide info view settings must also be selected. For further details on these, see section 19.1.1 and section 12.2.1.
  - **Low.** Hides trace data, quality scores and puts the reads’ annotations on the sequence. The editing functions available when right-clicking on a nucleotide with compactness set to Low is shown in figure 19.15.
Figure 19.14: Settings in the side panel allow customization of the view of read mappings and contigs from assemblies.

Figure 19.15: An example showing the low compactness setting.

- **Medium.** The labels of the reads and their annotations are hidden, and reads are shown as lines. The residues of the reads cannot be seen, even when zoomed in 100%.

- **Compact.** Like Medium but with less space between the reads.

- **Packed.** This uses all the horizontal space available for displaying the reads (figure 19.16). This differs from the other settings, which stack all reads vertically. When zoomed in to 100%, the individual residues are visible. When zoomed out, reads are represented as lines. Packed mode is useful when viewing large amounts of data, but some functionality is not available. For example, the read mapping cannot be edited, portions cannot be selected, and color coding changes are not possible.
• **Gather sequences at top.** When selected, the sequence reads contributing to the mapping at that position are placed right below the reference. This setting has no effect when the compactness level is Packed.

• **Show sequence ends.** When selected, trimmed regions are shown (faded traces and residues). Trimmed regions do not contribute to the mapping or contig.

• **Show mismatches.** When selected and when the compactness is set to Packed, mismatches that do not match the reference at that position are highlighted by coloring them according to the Rasmol color scheme. Reads with mismatches are floated to the top of the view.

• **Show strands of paired reads.** When the compactness is set to Packed, display each member of a read pair in full and color them according to direction. This is particularly useful for reviewing overlap regions in overlapping read pairs.

• **Packed read height.** When the compactness is set to “Packed”, select a height for the visible reads.

When there are more reads than the height specified, an overflow graph is displayed that uses the same colors as the sequences. Mismatches in reads are shown as narrow vertical lines, using colors representing the mismatching residue. Horizontal line colors correspond to those used for highlighting mismatches in the sequences (red = A, blue = C, yellow = G, and green = T). For example, a red line with half the height of the blue part of the overflow graph represents a mismatching “A” in half of the paired reads at that particular position.

• **Find Conflict.** Clicking this button selects the next position where there is an conflict. Mismatching residues are colored using the default color settings. You can also click on the Space bar of your keyboard to find the next conflict.

• **Low coverage threshold.** All regions with coverage up to and including this value are considered low coverage. Clicking the ‘Find low coverage’ button selects the next region in the read mapping with low coverage.

**Sequence layout.** There is one parameter in this section in addition to those described in section 12.2.1.
• **Matching residues as dots.** When selected, matching residues are presented as dots instead of as letters.

**Annotation Layout and Annotation Types** See section 12.3.1.

**Residue coloring.** There is one parameter in this section in addition to those described in section 12.2.1.

• **Sequence colors.** This setting controls the coloring of sequences when working in most compactness modes. The exception is Packed mode, where colors are controlled with settings under the "Match coloring" tab, described below.
  - **Main.** The color of the consensus and reference sequence. Black by default.
  - **Forward.** The color of forward reads. Green by default.
  - **Reverse.** The color of reverse reads. Red by default.
  - **Paired.** The color of read pairs. Blue by default. Reads from **broken pairs** are colored according to their orientation (forward or reverse) or as a non-specific match, but with a darker hue than the color of ordinary reads.
  - **Non-specific matches.** When a read would have matched equally well another place in the mapping, it is considered a non-specific match and is colored yellow by default. Coloring to indicate a non-specific match overrules other coloring. For mappings with several reference sequences, a read is considered a non-specific match if it matches more than once across all the contigs/references.

Colors can be adjusted by clicking on an individual color and selecting from the palette presented.

**Alignment info.** There are several parameters in this section in addition to the ones described in section 14.2.

• **Coverage:** Shows how many reads are contributing information to a given position in the read mapping. The level of coverage is relative to the overall number of reads.
• **Paired distance:** Plots the distance between the members of paired reads.
• **Single paired reads:** Plots the percentage of reads marked as single paired reads (when only one of the reads in a pair matches).
• **Non-specific matches:** Plots the percentage of reads that also match other places.
• **Non-perfect matches:** Plots the percentage of reads that do not match perfectly.
• **Spliced matches:** Plots the percentage of reads that are spliced.

Options for these parameters are:

• **Foreground color.** Colors the residues 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 residues using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage.
• **Graph.** Read coverage is displayed as a graph (Learn how to export the data behind the graph in section 7.4).
  - **Height.** Specifies the height of the graph.
- **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
- **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. If a Color bar is chosen, the color box is replaced by a gradient color box as described under Foreground color.

**Nucleotide info** See section 12.2.1

**Match coloring** Coloring of the mapped reads when the Packed compactness option is selected.

**Find** See section 12.2.1

**Text format** See section 12.2.1

19.7.2 Editing a contig or read mapping

When editing contigs and read mappings, you are typically interested in confirming or changing single bases.

To do this:

**Select the base to edit | Type the desired base**

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 ( )**

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 19.12).

**Note:** For contigs or mappings with more than 1,000 reads, you can only do single-residue replacements. When the compactness is **Packed**, you cannot edit any of the reads.

All changes are recorded in the history of the element (see section 2.5).

19.7.3 Sorting reads

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.

### 19.7.4 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 (red) are colored red.

- **Resolved.** Both the annotation and the corresponding row in the Table (green) 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.**

### 19.7.5 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 for Primer design (for example).

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 (save) or by dragging it to the Navigation Area.

### 19.7.6 Extract reads from a mapping

Note that the functionalities described in this page are valid for read mappings. For similar functionalities on tracks, see section ??.
Extract from Selection  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 for a detailed description of the different modes) and select your region of interest in your mapping, then right-click on the reference sequence or on the consensus sequence of the mapping (figure 19.17).

![Figure 19.17: Extracting parts of a mapping using the right-click menu available when clicking on a selected portion of the consensus sequence.](image)

When you choose the Extract from Selection option you are presented by the dialog shown in figure 19.18.

![Figure 19.18: Selecting the reads to include.](image)

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 16.2)

You will now have all paired reads from the original mapping in a list.

**Extract Sequences** When right-clicking on the sequences (as opposed to the consensus sequence), the menu to the right of figure 19.19 is available, and allows you to Extract Sequences from the mapping as explained in section 16.2. As opposed to the Extract from Selection tool, the Extract sequences will include all reads, not only the ones covered by the selection.
19.7.7 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 (表) icon at the bottom of the view.

This will display a new view of the conflicts as shown in figure 19.20.

The table has the following columns:

- **Reference position.** The position of the conflict measured from the starting point of the reference sequence.
• **Consensus position.** The position of the conflict measured from the starting point of the consensus sequence.

• **Consensus residue.** The consensus’s residue at this position. The residue can be edited in the graphical view, as described above.

• **Other residues.** Lists the residues of the reads. Inside the brackets, you can see the number of reads having this residue at this position. In the example in figure 19.20, 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 19.20). The comments are saved when you Save.

By clicking a row in the table, the corresponding position is highlighted in the graphical view. Clicking the rows of the table is another way of navigating the contig 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.

### 19.8 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. You can also right-click in the empty white area of the contig and choose to Reassemble contig.

This opens a dialog as shown in figure 19.21.

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 19.3. 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 19.4.
When you click Finish, a new contig is created, so you do not lose the information in the old contig.

19.9 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 19.2).

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 ( ) | Call Secondary Peaks ( )**

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 19.22.

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.
- **Peak slope stringency.** Control how pronounced each nucleotide peak must be. Decreasing this will detect more peaks. Increasing it will detect fewer.
• **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 an 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.

### 19.10 Extract Consensus Sequence

Using the **Extract Consensus Sequence** tool, a consensus sequence can be extracted from read mappings and from nucleotide BLAST results.

**Note:** Consensus sequences can also be extracted when viewing a read mapping by right-clicking on the name of the consensus or reference sequence, or a selection of the reference sequence, and selecting the option **Extract New Consensus Sequence** from the menu that appears. The same option is available from the graphical view of BLAST results when right-clicking on a selection of the subject sequence.

To start the **Extract Consensus Sequence** tool, go to:

**Toolbox | Sanger Sequencing Analysis | Extract Consensus Sequence**

In the first step, select the read mappings or nucleotide BLAST results to work with.

In the next step, options affecting how the consensus sequence is determined are configured (see figure 19.23).

#### Handling low coverage regions

The first step is to define a **low coverage threshold**. Consensus sequence is not generated for reference positions with coverage at or below the threshold specified.

The default value is 0, which means that a reference base is considered to have low coverage when no reads cover this position. Using this threshold, if just a single read covered a particular
position, only that read would contribute to the consensus at that position. Setting a higher threshold gives more confidence in the consensus sequence produced.

There are several options for how low coverage regions should be handled:

- **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 sequences 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 simply adds 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 uses 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).

**Handling conflicts**

Settings are provided in the lower part of the wizard for configuring how conflicts or disagreement between the reads should be handled when building a consensus sequence in regions with adequate coverage.

- **Vote** When reads disagree at a given position, the base present in the majority of the reads at that position is used for the consensus.
When choosing between symbols, we choose in the order A - C - G - T.

Ambiguous symbols cannot be chosen.

If the Use quality score option is also selected, quality scores are used to decide the base to use for the consensus sequence, rather than the number of reads. The quality scores for each base at a given position in the mapping are summed, and the base with the highest total quality score at a given position is used in the consensus. If two bases have the same total quality score at a location, we follow the order of preference listed above.

Information about biological heterozygous variation in the data is lost when the Vote option is used. For example, in 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.

- Insert ambiguity codes When reads disagree at a given position, an ambiguity code representing the bases at that position is used in the consensus. (The IUPAC ambiguity codes used can be found in Appendix G and F.) Unlike the Vote option, some level of information about biological heterozygous variation in the data is retained using this option.

To avoid the situation where a different base in a single read could lead to an ambiguity code in the consensus sequence, the following options can be configured:

- Noise threshold The percentage of reads where a base must be present at given position for that base to contribute to an ambiguity code. The default value is 0.1, i.e. for a base to contribute to an ambiguity code, it must be present in at least 10% of the reads at that position.

- Minimum nucleotide count The minimum number of reads a particular base must be present in, at a given position, for that base to contribute to the consensus.

If no nucleotide passes these two thresholds at a given position, that position is omitted from the consensus sequence.

If the Use quality score option is also selected, summed quality scores are used, instead of numbers of reads for conflict handling. To contribute to an ambiguity code, the summed quality scores for bases at a given position must pass the noise threshold.

In the next step, output options are configured (figure 19.24).

**Consensus annotations**

Annotations can be added to the consensus sequence, providing information about resolved conflicts, gaps relative to the reference (deletions) and low coverage regions (if the option to split the consensus sequence was not selected). Note that for large data sets, many such annotations may be generated, which will take more time and take up more disk space.

For stand-alone read mappings, it is possible to transfer existing annotations to the consensus sequence. Since the consensus sequence produced may be broken up, the annotations will also be broken up, and thus may not 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.

Quality scores on the consensus sequence

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 $X$. Let $Q = Z - (Y - Z)$, then we will assign $X$ the quality score of $q$ where

$$ q = \begin{cases} 
64 & \text{if } Q > 64 \\
0 & \text{if } Q < 0 \\
Q & \text{otherwise} 
\end{cases} $$

19.11 Combine Reports

The Combine Reports tool makes it easy to get a cross-sample overview by summarizing reports from multiple samples. The tool takes in multiple reports and generates a single report containing summaries and other selected information from the original reports with outliers highlighted.

This tool supports reports generated by the Trim Sequences tool. If other report types are selected as input, information from these will not appear in the combined report. If none of the

---

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 $Y = Z$. 

---
Creating a combined report

To create a combined report, go to:

Toolbox | Sanger Sequencing Analysis ( ) Combine Reports ( )

In the dialog that opens, select the reports to be combined (figure 19.25).

In the next dialog, configuration options are presented, as shown in figure 19.26:

The configuration options are:

- **Show summary information as plots** Enable this to display information as box plots instead of tables, if possible for a given data value.

- **Include tables for outliers** Enable this to add a table after each summary table or box plot, containing the samples that are outliers for that data type, if possible for a given data value.

Note: these configuration options are most applicable to reports produced by the CLC Genomics Workbench. They are not relevant when working with reports generated by the Trim Sequences tool. Complete trimming information is included in the combined report for all sequences.

19.11.1 Combine Reports output

The **Combine Reports** tool generates a single report, containing summaries and other selected information from the reports provided as input.
The report is divided into one or more sections depending on the input report type.

By default, summary information is displayed in table format. Each table presents a summary of the corresponding information from the input reports.
Chapter 20

Primers and probes

Contents

20.1 Primer design - an introduction ................................................. 437
  20.1.1 General concept .......................................................... 437
  20.1.2 Scoring primers .......................................................... 439
20.2 Setting parameters for primers and probes .............................. 439
  20.2.1 Primer Parameters ....................................................... 439
20.3 Graphical display of primer information ................................... 441
  20.3.1 Compact information mode ........................................... 441
  20.3.2 Detailed information mode ........................................... 442
20.4 Output from primer design .................................................... 443
20.5 Standard PCR ..................................................................... 444
  20.5.1 When a single primer region is defined .............................. 445
  20.5.2 When both forward and reverse regions are defined ............. 446
  20.5.3 Standard PCR output table ............................................ 447
20.6 Nested PCR ...................................................................... 448
20.7 TaqMan .......................................................................... 449
20.8 Sequencing primers .............................................................. 451
20.9 Alignment-based primer and probe design .............................. 452
  20.9.1 Specific options for alignment-based primer and probe design . 453
  20.9.2 Alignment based design of PCR primers ............................ 454
  20.9.3 Alignment-based TaqMan probe design ............................. 455
20.10 Analyze primer properties ...................................................... 457
20.11 Find binding sites and create fragments ................................. 458
  20.11.1 Binding parameters ..................................................... 459
  20.11.2 Results - binding sites and fragments ............................. 460
20.12 Order primers ................................................................... 462

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.

20.1 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 20.1), the basic options for viewing the template sequence are the same as for the standard sequence view (see section 12.2 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 20.1: The initial view of the sequence used for primer design.

20.1.1 General concept

The concept of the primer view is that the user first chooses the desired reaction type for the session in the Primer Parameters preference group, e.g. Standard PCR. Reflecting the choice of reaction type, it is now possibly to select one or more regions on the sequence and to use the right-click mouse menu to designate these as primer or probe regions (see figure 20.2).

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 showed using a compact mode but the user can change to a more detailed mode in the Primer information preference group.

The number of information lines reflects the chosen length interval for primers and probes. In the compact information mode one line is shown for every possible primer-length and each of these lines contain information regarding all possible primers of the given length. At each potential primer starting position, a circular information point is shown which indicates whether the primer fulfills the requirements set in the primer parameters preference group. A green circle indicates a primer which 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.
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.
20.1.2 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.

20.2 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 20.3).

![Figure 20.3: The two groups of primer parameters (in the program, the Primer information group is listed below the other group).](image)

20.2.1 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\(^{2+}\)]) in units of millimoles (mM)
    
    - **dNTP concentration.** Specifies the combined concentration of all deoxynucleotide triphosphates in units of millimoles (mM)
    
    - **DMSO concentration.** Specifies the concentration of dimethyl sulfoxide in units of volume percent (vol.\%)
  
  - **GC content.** Determines the interval of CG content (% C and G nucleotides in the primer) within which primers must lie by setting a maximum and a minimum GC content.
  
  - **Self annealing.** Determines the maximum self annealing value of all primers and probes. This determines the amount of base-pairing allowed between two copies of the same molecule. The self annealing score is measured in number of hydrogen bonds between two copies of primer molecules, with A-T base pairs contributing 2 hydrogen bonds and G-C base pairs contributing 3 hydrogen bonds.
  
  - **Self end annealing.** Determines the maximum self end annealing value of all primers and probes. This determines the number of consecutive base pairs allowed between the 3’ end of one primer and another copy of that primer. This score is calculated in number of hydrogen bonds (the example below has a score of 4 - derived from 2 A-T base pairs each with 2 hydrogen bonds).

  \[
  \begin{align*}
  \text{AATTCCCTACAATCCCCAAA} & \quad || \\
  \text{AAACCCTAACATCCCTTA} \end{align*}
  \]
- **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

### 20.3 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 20.3).

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.

#### 20.3.1 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 20.4).
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.

### 20.3.2 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 20.5).

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:

- 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.

### 20.4 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 20.6).

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 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*.

### 20.5 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 20.3) and setting the desired parameter values in the Primer Parameters preference group, the Calculate button will activate the primer design algorithm.

### 20.5.1 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 20.7).

![Figure 20.7: Calculation dialog for PCR primers when only a single primer region has been defined.](image)

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.

20.5.2 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 20.8).

Figure 20.8: 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 20.5.1). 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.

**20.5.3 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
20.6 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 20.3) 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 20.9).

Figure 20.9: Calculation dialog for nested primers.
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 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.

### 20.7 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 20.3) 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 20.10) which is similar to the Nested PCR dialog described above (see section 20.6).

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.

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.

### 20.8 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 20.3) 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 20.11).

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 20.5 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.

### 20.9 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 (primers) | Design Primers (design)
```

Or if the alignment is already open, click Primer Designer (design) in the lower left part of the view.

In the alignment primer view (see figure 20.12), the basic options for viewing the template alignment are the same as for the standard view of alignments (see section 14 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.
20.9.1 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.

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 20.12):

- 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 20.13):

- 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.

### 20.9.2 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 $4 \times 4 \times 2 = 32$ 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 20.14.

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.

![Calculation parameters](image)

Figure 20.14: Calculation dialog shown when designing alignment based PCR primers.

### 20.9.3 Alignment-based TaqMan probe design

**CLC Main Workbench** allows the user to design solutions for TaqMan quantitative PCR which consist of four oligos: a general primer pair which will amplify all sequences in the alignment, a specific TaqMan probe which will match the group of *included* sequences but *not* match the *excluded* sequences and a specific TaqMan probe which will match the group of *excluded*
sequences but not match the included sequences. As above, the selection boxes are used to indicate the status of a sequence, if the box is checked the sequence belongs to the included sequences, if not, it belongs to the excluded sequences. We use the terms included and excluded here to be consistent with the section above although a probe solution is presented for both groups. In TaqMan mode, primers are not allowed degeneracy or mismatches to any template sequence in the alignment, variation is only allowed/required in the TaqMan probes.

Pushing the **Calculate** button will cause the dialog shown in figure 20.15 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 20.15: Calculation dialog shown when designing alignment based TaqMan probes.](image)

### 20.10 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 20.16:

![Figure 20.16: The parameters for analyzing primer properties.](image)
In the *Concentrations* panel a number of parameters can be specified concerning the reaction mixture and which influence melting temperatures

- **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM)

- **Salt concentration.** Specifies the concentration of monovalent cations ([Na$^+$], [K$^+$] and equivalents) in units of millimoles (mM)

In the *Template panel* the sequences of the chosen primer and the template sequence are shown. The template sequence is as default set to the reverse complement of the primer sequence i.e. as perfectly base-pairing. However, it is possible to edit the template to introduce mismatches which may affect the melting temperature. At each side of the template sequence a text field is shown. Here, the dangling ends of the template sequence can be specified. These may have an important affect on the melting temperature [Bommarito et al., 2000]

Click **Finish** to start the tool. The result is shown in figure 20.17:

![Figure 20.17: Properties of a primer.](image)

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 20.5.3.

### 20.11 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.

### 20.11.1 Binding parameters

This opens the dialog displayed in figure 20.18:

![Figure 20.18: Search parameters for finding primer binding sites.](image)

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)
20.11.2 Results - binding sites and fragments

Specify the output options as shown in figure 20.19:

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 20.20.

The annotation has the following information:

- **Sequence of the primer.** Positions with mismatches will be in lower-case (see the fourth position in figure 20.20 where the primer has an *a* and the template sequence has a *T*).
• **Number of mismatches.**

• **Number of other hits on the same sequence.** This number can be useful to check specificity of the primer.

• **Binding region.** This region ends with the 3’ exact match and is simply the primer length upstream. This means that if you have 5’ extensions to the primer, part of the binding region covers sequence that will actually not be annealed to the primer.

An example of the **primer binding site table** is shown in figure 20.21.

![Figure 20.21: A table showing all binding sites.](image)

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 20.5.3. You can use this table to browse the binding sites. If you make a split view of the table and the sequence (see section 2.1.4), 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 20.22.

![Figure 20.22: A table showing all possible fragments of the specified size.](image)

The table first lists the names of the forward and reverse primers, then the length of the fragment and the region. The last column tells if there are other possible fragments fulfilling the length
criteria on this sequence. This information can be used to check for competing products in the PCR. In the Side Panel you can show information about melting temperature for the primers as well as the difference between melting temperatures.

You can use this table to browse the fragment regions. If you make a split view of the table and the sequence (see section 2.1.4), 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 20.23.

![Figure 20.23: Right-clicking a fragment allows you to annotate the region on the input sequence or open the fragment as a new sequence.](image)

This will put a PCR fragment annotations on the input sequence covering the region specified in the table. As you can see from figure 20.23, 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 21.3).

### 20.12 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 (opens window) | Order Primers (opens window)](image)

This opens a dialog where you can choose primers to generate a textual representation of the primers (see figure 20.24).

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 20.24: A primer order for 4 primers.
Chapter 21

Cloning and restriction sites

Contents

21.1 Restriction site analyses ........................................ 464
  21.1.1 Dynamic restriction sites ................................... 465
  21.1.2 Restriction Site Analysis .................................... 470
  21.1.3 Insert restriction site ....................................... 473
21.2 Restriction enzyme lists .......................................... 473
21.3 Restriction Based Cloning ......................................... 475
  21.3.1 Introduction to the Cloning Editor ......................... 476
  21.3.2 The restriction cloning workflow ............................ 477
  21.3.3 Manual cloning ............................................ 479
21.4 Homology Based Cloning .......................................... 483
  21.4.1 Working with homology based cloning ...................... 484
  21.4.2 Adjust the homology based cloning design ............... 485
  21.4.3 Homology Based Cloning outputs .......................... 487
  21.4.4 Detailed description of the Homology Based Cloning wizard ........................................ 488
  21.4.5 Working with mutations ................................... 492
21.5 Gateway cloning .................................................. 492
  21.5.1 Add attB sites ............................................. 493
  21.5.2 Create entry clones (BP) .................................. 495
  21.5.3 Create expression clones (LR) ............................. 497
21.6 Gel electrophoresis ................................................ 498
  21.6.1 Gel view ................................................ 499

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.

21.1 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 Restriction Sites Analysis 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.

21.1.1 Dynamic restriction sites

If you open a sequence or a sequence list, 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 21.1): check the "Show" option on top of the Restriction sites section, then specify the enzymes that should be displayed.

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 above the list of restriction enzymes.

There is also an option to specify how the Labels should be shown:

- **No labels.** This will just display the cut site with no information about the name of the enzyme. Placing the mouse button on the cut site will reveal this information as a tool tip.

- **Flag.** This will place a flag just above the sequence with the enzyme name (see an example in figure 21.2). 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.
• **Radial.** This option is only available in the circular view. It will place the restriction site labels as close to the cut site as possible (see an example in figure 21.3).

![Figure 21.3: Restriction site labels in radial layout.](image)

• **Stacked.** This is similar to the flag option for linear sequence views, but it will stack the labels so that all enzymes are shown. For circular views, it will align all the labels on each side of the circle. This can be useful for clearly seeing the order of the cut sites when they are located closely together (see an example in figure 21.4).

![Figure 21.4: Restriction site labels stacked.](image)

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 21.5).

![Figure 21.5: Buttons to sort restriction enzymes.](image)

• **Sort enzymes alphabetically** ($A_a$). Clicking this button will sort the list of enzymes alphabetically.

• **Sort enzymes by number of restriction sites** ($T^k$). 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** (T). 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 21.6.

![Figure 21.6: Adding or removing enzymes from the Side Panel.](image)

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 (T). 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 21.7). You can also open a view of an enzyme list saved in the Navigation Area.

**Figure 21.7:** 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 on **Finish**, the enzymes are added to the Side Panel and the cut sites are shown on the sequence. You can save the settings in the Side Panel, including the enzymes just added, as described in section 4.6).

### 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 21.1.1). 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 21.8), 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 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 (⌘ D) to find enzymes producing a compatible overhang (figure 21.9).
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.

This functionality does not work for enzymes where the cut site is located outside the recognition site.

### 21.1.2 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 Restriction Site Analysis](image)

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 21.1.1).

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 21.10).

![Figure 21.10: Selecting number of cut sites.](image)

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 21.11) 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 21.12) and save the annotations for later use.

Create restriction map. When a restriction map is created, it can be shown in three different ways:

- As a table of restriction sites as shown in figure 21.13. 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.

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 21.14). Click the Fragments button at the bottom of the view.

![Figure 21.14: The result of the restriction analysis shown as table of fragments.](image)

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 21.48). For more information about gel electrophoresis, see section 21.6.

21.1.3 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 21.15.

![Figure 21.15: Inserting a restriction site and potentially a recognition sequence.](image)

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.

21.2 Restriction enzyme lists

CLC Main Workbench includes 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 form 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](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 21.16

![Figure 21.16: Choosing enzymes for the new enzyme list.](image)

Choose which enzyme you want to include in the new enzyme list (see section 21.1.1), and click **Finish** to open the enzyme list.

View and modify enzyme list  

An enzyme list is shown in figure 21.17. 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.

If you wish to remove or add enzymes, click the **Add/Remove Enzymes** button at the bottom of the view. This will present the same dialog as shown in figure 21.16 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.
21.3 Restriction Based Cloning

The in silico cloning process in CLC Main Workbench begins with the Restriction Based Cloning tool:

**Cloning** (inations) | **Restriction Based Cloning** (inations)

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 21.18).

[Figure 21.18: Selecting the sequences containing the fragments you want to clone and the vector.]

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 (inations) at the bottom of the view (figure 21.19).

If you later in the process need additional sequences, right-click anywhere on the empty white
21.3.1 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 12.2). 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 opened in the following ways:

- Click on the Cloning Editor icon in the view area when a sequence list has been opened in the sequence list editor.
• Create a new cloning experiment using the **Restriction Based Cloning** (🛠️) action from the toolbox. This tool collects a set of existing sequences and creates a new sequence list.

The Cloning Editor can be used in two ways:

- **Cloning mode** Opened when one of the sequences has been selected as 'Vector'. In this mode, you can apply 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 order and direction of the inserted fragments can be adjusted prior to adjusting the overhangs to match the cloning conditions.

- **Stitch mode** If no sequence has been selected as 'Vector', a number of fragments (either full sequences or cuttings) can be selected from the cloning experiment. These can then be stitched together into a single new sequence. In the stitching adapter dialog, the order and direction of the fragments can be adjusted prior to adjusting the overhangs to match the stitch conditions.

### 21.3.2 The restriction cloning workflow

The restriction cloning workflow consists of 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 21.1.1). 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 21.20).

   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.

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 21.21).

   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 21.22.

This dialog visualizes the details of the insertion. The vector sequence is on each side
Figure 21.22: Showing the insertion point of the vector.

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.

21.3.3 Manual cloning

Cloning steps can be done manually. For this, create a sequence list using Restriction Based Cloning and use the cloning actions provided via right-click menus. The general workflow is the same as that described in section 21.3.2, but carried out manually.

Manipulate the whole sequence

Right-click the sequence label to the left to see the menu shown in figure 21.23.

- **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 21.6

- **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 21.24.
• **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 (X).** Cleaves the sequence before the selection and will result in two smaller fragments.

• **Cut Sequence After Selection (X).** Cleaves the sequence after the selection and will result in two smaller fragments.

• **Make Positive Strand Single Stranded (X).** Makes the positive strand of the selected region single stranded.

• **Make Negative Strand Single Stranded (X).** Makes the negative strand of the selected region single stranded.

• **Make Double Stranded (X).** 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 (C).** Copies the selected region to the clipboard, which will enable it for use in other programs.

• **Open Selection in New View (O).** Opens the selected region in the normal sequence view.

• **Edit Selection (E).** Opens a dialog box in which is it possible to edit the selected residues.

• **Delete Selection (Del).** 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 21.1.3).

• **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 21.25).

![Figure 21.25: Select a sequence for insertion.](image)

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 21.3.2.

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). 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 21.26).

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.

21.4 Homology Based Cloning

Using Homology Based Cloning, cloning experiments can be designed for methods such as Takara In-Fusion® and other ligation independent cloning techniques and Gibson Assembly®, where sequences with homologous ends need to be assembled.

The tool generates primers with overhangs so that appropriate homologous sequence is added to fragments that should be assembled. Subsequently, primers and overhangs can be inspected
and adjusted.

Running the Homology Based Cloning tool

To run the tool, go to:

Cloning ( ) Homology Based Cloning ( )

In the first dialog, select the vector and all the DNA fragments that should be assembled in the cloning reaction (figure 21.28).

The first sequence used as input, will be considered the vector, but the order of sequences can be adjusted in the following wizard step.

You can select 1-50 individual sequences with a maximal combined length of 100,000 base pairs as input to the tool. However, the number of sequences selected should be in accordance with the relevant laboratory protocol.

Note that it is also possible to use an assembled vector that was previously generated by Homology Based Cloning as input. In that case, previously designed primers and overhangs will be shown in the following wizard step.

![Figure 21.28: Select the vector and fragments that should be assembled in the homology based cloning reaction.](image)

Press Next to open the wizard allowing you to inspect and adjust primers and overhangs.

21.4.1 Working with homology based cloning

This section gives a quick overview of the Homology Based Cloning wizard. For a full description, see section 21.4.4.

General options

General options are at the top of the wizard. These include the position of the insertion site in the vector, the maximum primer and overhang lengths as well as option to set the Tm and overhang length for all primers at once. There is also a diagram of the vector including the inserts, where each sequence has a different colour (figure 21.29).

Sequences

Each sequence is displayed individually, with a coloured bar to the left and a vertical scroll bar at the bottom. The top sequence is the vector, with the insert sequences displayed further down.
Figure 21.29: The top section of the wizard contains general options.

The order of the sequences reflects how they will be assembled into the vector, and the overhangs on the primers support this assembly order.

Vector, inserts, primers and overhangs are color coded (figure 21.30):

- **Black** Vector or insert sequence
- **Grey** Vector or insert sequence not included in the PCR product
- **Brown** Primer sequence
- **Pink** Overhang sequence
- **Blue** Added bases that are inserted between primer and overhang

For each sequence, you can adjust primer and overhang lengths and add bases between primers and overhangs.

The vector sequence is considered circular and primers are depicted as pointing away from each other in order to amplify the circular sequence. Inserts are considered linear, and primers are placed at the ends of the insert sequence pointing towards each other in order to amplify the linear sequence (figure 21.30).

### 21.4.2 Adjust the homology based cloning design

This section describes how primers and overhangs can be adjusted and inspected.

The overhangs added by **Homology Based Cloning** are 20 base pairs long and are added to vector or insert primers:

- If one insert is assembled into a vector < 8 kb in length, overhangs are added to the vector primers.
- If one insert is assembled into a vector > 8 kb in length, overhangs are added to the insert primers.
- If more inserts are assembled into a vector, the overhangs are added to insert primers.

Primer and overhang lengths should be adjusted, according to the cloning kit used.
Figure 21.30: Top: The vector sequence and primers with overhangs. The grey sequence between the primers is not included in the PCR product. Bottom: An insert sequence and primers with overhangs.

**Change the insertion site**

Change the insertion site in the vector by typing a position or a range of positions directly into **Insertion site** text field. Alternatively, choose the start, end or complete range of an annotation from the drop down menu (figure 21.31). If you specify a range of bases, primers will be placed so that bases are replaced by the insert(s).

Figure 21.31: Choose an insertion site from the drop down menu or type position(s) directly in the Insertion site text field.

**Change the assembly order**

Use the arrows to the left of the sequence names to move a sequence up or down in the order. The sequence of primer overhangs is automatically updated to reflect the new order. The length of overhangs is not changed.

**Adjust the length of primers and overhangs**

You can adjust primers and overhangs using the following options:
To adjust all primers at once, change the **Primer Tm** in the top section and press **Calculate primers** (figure 21.29). This will update primers on all sequences.

To adjust all overhangs at once, change the **Overhang length** in the top section and press **Set Overhang Lengths** (figure 21.29). This will update overhangs on all sequences.

To adjust the length of individual primers and overhangs use the **Primer length** and **Overhang length** options available for the forward and reverse primer on each sequence. You can also extend or shorten the the primer and overhang sequences by dragging the arrow symbols at the ends of the primers and overhangs (figure 21.30).

### Insert additional bases

Insert additional bases between the primer and the overhang by typing directly in to the **Added bases** text fields. You can also choose the sequence of a restriction site from the drop down menu.

### Inspect primer pairs

The designed primer pairs can be closely inspected in the table that opens when pressing **Open Primer Pairs Table**. The table contains information about secondary structure for each primer pair, both with and without overhang. For a description of each of the columns in the Primer Pairs table see section 20.5.3. Close the table before returning to the wizard to make further adjustments or complete the design.

The outputs created by **Homology Based Cloning** are described in section 21.4.3.

### 21.4.3 Homology Based Cloning outputs

**Homology Based Cloning** can create the following outputs:

- **Report** A report containing the following sections:
  - **Summary** Contains the number of fragments and primers used in the cloning reaction as well as their lengths and any warnings.
  - **Fragments** Lists the vector and fragments used in the cloning reaction.
  - **Warnings** Lists the warnings given for primer pairs.
  - **Primer pairs** Lists fragments for which primers were designed together with pair annealing and pair end annealign values for the primer pairs. See section 20.5.3 for information about annealing values.
  - **Primers** Lists individual primers and their sequence. Primer sequence is written with capital letters, whereas added bases and overhangs are in lowercase.
  - **Primer parts** Lists full and subparts of designed primers with characteristics such as length and G/C content. The following terms are used:
    - **Full** The full primer including overhang and added bases.
    - **Anneal** The part of the primer annealing to the original fragment (primer without overhang or any added bases).
• **Overhang** The overhang added to primers to generate homologous sequence between fragments.
• **Added** The bases inserted between primer and overhang.

- **Assembled vector** The vector as it will appear after all fragments have been assembled. The assembled vector will be annotated with the positions of primers, added bases and overhangs as well as with inserts and vector sequence. When the vector is opened, you can select which annotations should be shown on the sequence in the side panel under Annotation types.

  Note: the assembled vector can be used as input to **Homology Based Cloning** if you wish to adjust a previous design.

- **Primers sequence list** A sequence list containing the designed primers. The primers are annotated with primer, added bases and overhang, where primer is the part of the sequence that originally aligned to the insert or vector that was amplified.

- **PCR fragments sequence list** The PCR fragments generated from input sequences and designed primers including additional bases and overhangs.

- **Primer pairs table** A table providing information about melting temperatures, secondary structure, etc., for primer pairs with and without overhangs. For a description of each of the columns in the Primer Pairs table, see section **20.5.3**.

### 21.4.4 Detailed description of the Homology Based Cloning wizard

This section contains a detailed description of the **Homology Based Cloning** options (figure 21.32).

- **Insertion site** The position where fragments will be inserted in the vector. You can type in a specific position or a range of positions. You can also choose the start, the end, or the entire span of an annotation on the vector using the drop down menu (figure 21.33).

  The primers designed to amplify the vector will be placed so that their 5’ ends are adjacent to the insertion site. If a range of positions are selected, the primers will be placed so that the selected positions are not included in the PCR product. When the insertion site is changed, the vector primers in the view below are updated accordingly.

  Insertion site examples:
  - 0 or 0*1 Assembles inserts into the vector between the last and the first base.
  - 1 or 1*2 Assembles inserts into the vector between the first and second base.
  - 1..10 Inserts replace bases 1-10 in the vector.
  - **Start of an annotation** Assembles inserts into the vector before the first base in the annotated region.
  - **Span of an annotation** Inserts replace all bases in the annotated region.
  - **End of an annotation** Assembles inserts into the vector after the last base in the annotated region.
Figure 21.32: General options for the cloning experiment are provided at the top of the wizard, followed by sections for the vector and insert sequences, where many options relevant to the cloning experiment can be adjusted.

- **Maximum primer length** The maximum length that primers for vectors and inserts can be. This is reflected in the number of nucleotides visible for each sequence in the views below.

- **Maximum overhang length** The maximum length that overhangs for vectors and inserts can be.
Figure 21.33: Specify the insertion site in the vector. Here the entire Lac-operon has been selected from the drop down menu. Notice that when a span of bases are chosen as insertion site, the vector sequence between the primers is grey and not included in the PCR product.

- **Font size** The font size to use for vector and insert sequences, and for primers and overhangs.

- **Primer Tm** The primer melting temperature. This value does not take into account any added bases or overhangs. Click on **Calculate primers** to update all primers after changing this value.

- **Overhang length** The length of the overhang added to primers not including added bases. Click on **Set Overhang Lengths** to update overhangs after changing this value.

- **Open Primer Pairs Table** Opens a table listing each of the primer pairs shown on the sequences below. The primer pairs table contains primer pairs, both with and without overhangs and added bases. It also provides information about melting temperatures, secondary structure, etc. For a description of each of the columns in the Primer Pairs table, see section 20.5.3.

- **Vector map** A vector map showing the assembled vector. Each original fragment has its own color that matches the side bars of the sequences in the views below. If you hover over the sequence of the vector or an insert, it will become bold in the vector map. If you hover over a primer, it will appear on the vector map. The fragments, but not the primers are drawn to scale.

- **Sequence Name: n (vector, circular)** and **Sequence Name: n (insert y, linear)**. The sequences identified as the vector and inserts.

- **Arrows to the left of Sequence Name** Change the order of the sequences in the list using the up and down arrows. Reverse complement the sequence using the horizontal arrows.

- **Primer length** and **Overhang length** The primer and overhang lengths for the forward and reverse primer, respectively. These lengths can be adjusted by typing new values into the
dialogs, or by using the up and down arrows to the right of the dialogs. Changes to the lengths are immediately updated in the sequence view below. The Tm and primer pair annealing alignment are also updated.

- **Tm**  The primer melting temperature. This value does not include overhangs or any added bases.

- **Primer pair annealing alignment**  Predicted primer-primer annealing of the forward and reverse primers. Overhangs and added bases are not included. The same plot is also available in the primer pairs table.

- **Added bases**  Insert additional bases between the primer and overhang. You can either type the bases directly into the dialog, or you can choose the sequence of a specific restriction enzyme from the drop down menu.

- **Sequence and primer views**  For each sequence included in the homology cloning reaction, you can see the part of the sequence that primers are designed to, as well as the primers and their overhangs. For the vector, the fragment is considered circular and the primers are placed pointing in opposite directions from the insertion site (figure 21.34). Inserts are considered linear and primers are placed at the ends (figure 21.35).

  Vector, inserts, primers and overhangs are color coded (figure 21.30):
  
  - **Black**  Vector or insert sequence
  - **Grey**  Vector or insert sequence not included in the PCR product
  - **Brown**  Primer sequence
  - **Pink**  Overhang sequence
  - **Blue**  Added bases that are inserted between primer and overhang

The overhang of a primer for a given sequence is identical to the sequence that it will be adjacent to in the assembled vector. Figures 21.34 and 21.35 show an example where the linear sequence can be inserted into the circular sequence. Pink overhang bases on the primers for the circular fragment are either the same sequence or complementary to the black sequence of the linear DNA fragment. Overhangs are designed to assemble the fragments in the order they appear in the wizard. In this example, two sequences are assembled, but more than two can be used for homology based cloning.

![Figure 21.34: Vector sequence and corresponding primers.](image)

![Figure 21.35: Insert sequence and corresponding primers.](image)

- **Warnings in sequence views**  A yellow or red exclamation mark next to the sequence name warns of any problems 21.36. Hover over the primer to get more information from the
tooltip or click on the warning to open a dialog showing the warning message. Examples of when warnings appear include:

- A primer does not have G/C content between 40 and 60%.
- Only one primer is designed for a given fragment.
- An insert with a length that is not a multiple of 3 is placed inside a vector region annotated as a CDS and so may disrupt the reading frame.

Figure 21.36: Hover over the yellow exclamation mark to see the warnings in the tooltip.

### 21.4.5 Working with mutations

Mutations can be introduced using PCR amplification with primers containing the mutations. This can be mimicked in the software:

- Introduce mutations manually in the sequences before running Homology Based Cloning. Place primers over mutated sites to ensure the mutations are included in the primer sequence.
- Run Homology Based Cloning using the original sequences, and then introduce mutations into the assembled vector. Re-run Homology Based Cloning using the vector containing the mutations.

### 21.5 Gateway cloning

*CLC Main Workbench* offers tools to perform *in silico* Gateway cloning (Thermo Fisher Scientific), including Multi-site Gateway cloning.

The three tools for doing Gateway cloning in the *CLC Main Workbench* mimic the procedure 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.

21.5.1 Add attB sites

The first step in the Gateway cloning process is to amplify the target sequence with primers including so-called attB sites:

This will open a dialog where you can select the input sequence, sequences, or sequence lists containing fewer than 10,000 sequences. Note that if a 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 21.37.

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 21.38). 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 21.5.1.

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.
Figure 21.38: 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.

In the next step, specify the length of the template-specific part of the primers as shown in figure 21.39.

Figure 21.39: Specifying the length of the template-specific part of the primers.

The 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 21.40).

The attB sites, the primer additions and the primer regions are annotated in the final result as shown in figure 21.41 (you may need to switch on the relevant annotation types to show the sites and primer additions).

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 21.5.2).

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 21.38 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:
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.

The attB site plus the Shine-Dalgarno primer addition is annotated.

In the table Multisite Gateway Cloning primer additions (see figure 21.42), 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).
- **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.

### 21.5.2 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 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** | **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 21.5.1). You can select more than one sequence as input, and the corresponding number of entry clones will be created.

In the following dialog (figure 21.43), 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 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 21.44.

Figure 21.44: The resulting entry vector opened in a circular view.

Note that the bi-product of the recombination is not part of the output.

21.5.3 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 \( \text{ccdB} \) gene. This gene must be present either as a \('\text{ccdB}'\) annotation, or as the exact sequence:

\[
\begin{align*}
\text{ATGCAGTTAAGGTTCATACCTATAAAAGAGAGGCCGGATGTATCAGAGTGATATT} \\
\text{ATTGACACGCCGGCGAAGGTAGTATCCCCCTGCGGACGGTGACGTCGATCCGATACGTCCTCCTC} \\
\text{CGTGAACCTTAACCGCGTGATCGCAGCTGGGATGACGATGACCGGTGCATCGGCTGACTTCCTCC} \\
\text{GTGCCGGGTCTCGGTATCGGGCGAAAGAGTGGCTGTGACCTCGCAGCGGCGAAATGACATTCAGGCGC} \\
\text{ATTAACCTGATGTTCTGCGGAATATAA}
\end{align*}
\]

If the \( \text{ccdB} \) gene is not present or if the sequence is not identical to the above, a solution is to simply add a \('\text{ccdB}'\) annotation. Select part of the vector sequence, right-click and choose ‘Add Annotation’. Name the annotation ‘\text{ccdB}’.

You can now start the tool:

Toolbox | Cloning ( ) | Gateway Cloning ( ) | Create Expression Clone ( )

In the first step, select one or more entry clones (see how to create an entry clone in section 21.5.2). If you wish to perform separate LR reactions with multiple entry clones, you should run the Create Expression Clone in batch mode (see section 9.3).
In the second step, select the destination vector that was previously saved in the Navigation Area (fig 21.45).

![Figure 21.45: Selecting one or more destination vectors.](image)

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](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 21.46.

![Figure 21.46: The resulting expression clone opened in a circular view.](image)

You can choose to create a sequence list with the bi-products as well.

### 21.6 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 a restriction map which can be shown as a gel (see section 21.1.2).

• 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 12.2.5).
  - Ordinary sequence view (see section 12.2).
  - Graphical view of sequence lists (see section 12.1).
  - Cloning editor (see section 21.3).
  - Primer designer (see section 20.3).

• **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 21.47).

![Figure 21.47: A sequence list shown as a gel.](image)

### 21.6.1 Gel view

In figure 21.48 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:
Figure 21.48: Five lanes showing fragments of five sequences cut with restriction enzymes.

- Fragment length
- Fragment region on the original sequence
- Enzymes cutting at the left and right ends, respectively

For gels comparing whole sequences, you will see the sequence name and the length of the sequence.

**Note!** You have to be in **Selection** or **Pan** mode in order to get this information.

It can be useful to add markers to the gel which enables you to compare the sizes of the bands. This is done by clicking **Show marker ladder** in the **Side Panel**.

Markers can be entered into the text field, separated by commas.

**Modifying the layout**  The background of the lane and the colors of the bands can be changed in the **Side Panel**. Click the colored box to display a dialog for picking a color. The slider **Scale band spread** can be used to adjust the effective time of separation on the gel, i.e. how much the bands will be spread over the lane. In a real electrophoresis experiment this property will be determined by several factors including time of separation, voltage and gel density.

You can also choose how many lanes should be displayed:

- **Sequences in separate lanes**. This simulates that a gel is run for each sequence.
- **All sequences in one lane**. This simulates that one gel is run for all sequences.

You can also modify the layout of the view by zooming in or out. Click **Zoom in** or **Zoom out** in the **Toolbar** and click the view.

Finally, you can modify the format of the text heading each lane in the **Text format preferences** in the **Side Panel**.
Chapter 22

RNA structure

Contents

22.1 RNA secondary structure prediction ................................................. 502
  22.1.1 Selecting sequences for prediction ........................................ 502
  22.1.2 Secondary structure prediction parameters ............................... 503
  22.1.3 Structure as annotation ..................................................... 507

22.2 View and edit secondary structures ........................................... 508
  22.2.1 Graphical view and editing of secondary structure .................... 508
  22.2.2 Tabular view of structures and energy contributions ............... 511
  22.2.3 Symbolic representation in sequence view ............................. 514
  22.2.4 Probability-based coloring ................................................. 515

22.3 Evaluate structure hypothesis ................................................ 516
  22.3.1 Selecting sequences for evaluation ...................................... 516
  22.3.2 Probabilities .............................................................. 517

22.4 Structure scanning plot ....................................................... 518
  22.4.1 Selecting sequences for scanning .................................... 518
  22.4.2 The structure scanning result ......................................... 519

22.5 Bioinformatics explained: RNA structure prediction by minimum free energy minimization ..................................................... 520
  22.5.1 The algorithm .............................................................. 521
  22.5.2 Structure elements and their energy contribution .................. 522

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

### 22.1 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 22.5.

In **CLC Main Workbench**, structures are predicted by a modified version of Professor Michael Zuker's 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](http://mfold.rna.albany.edu/?q=mfold/mfold-references).

#### 22.1.1 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 22.1.

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.
CHAPTER 22. RNA STRUCTURE

22.1.2 Secondary structure prediction parameters

Click Next to adjust secondary structure prediction parameters (figure 22.2).

Structure output

The predict secondary structure algorithm always calculates the minimum free energy structure of the input sequence. In addition to this, it is also possible to compute a sample of suboptimal structures by ticking the checkbox 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 22.3.

![Figure 22.3: The marginal base pair probability of all possible base pairs.](image)

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 22.2.3 and also in the secondary structure view. An example is shown in figure 22.4. Furthermore, the marginal probabilities are accessible from tooltips when hovering over the relevant parts of the structure.
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 Asymmetric 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 22.1.2).

To **force** two regions to form a stem, open a normal sequence view and:

1. 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 22.5).

   ![Figure 22.5: Force a stem of the selected bases.](image)

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:

1. 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 22.6).

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:
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 22.7.

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.

22.1.3 Structure as annotation

You can choose to add the elements of the best structure as annotations (see figure 22.8).

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 22.2.2.
22.2 View and edit secondary structures

When you predict RNA secondary structure (see section 22.1), 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 22.2. See an example in figure 22.8.
- Symbolic representation below the sequence (see section 22.2.3).
- A graphical view of the secondary structure (see section 22.2.1).
- 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 22.2.2.

22.2.1 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 22.9.

Like the normal sequence view, you can use Zoom in and Zoom out. Zooming in will reveal the residues of the structure as shown in figure 22.9. 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 12.2.1. 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 (Fig. 22.2). 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 22.2.2 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 22.10).
  
  - **Proportional.** Arc lengths are proportional to the number of residues (see figure 22.11). Nothing is done to prevent overlap.
  
  - **Even spread.** Stems are spread evenly around loops as shown in figure 22.12.

• **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 22.10: Auto layout. Overlaps are minimized.](image)

![Figure 22.11: Proportional layout. Length of the arc is proportional to the number of residues in the arc.](image)

![Figure 22.12: Even spread. Stems are spread evenly around loops.](image)

**Selecting and editing**

When you are in **Selection mode** (Fig. 22.3), 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 22.13).

![Figure 22.13: A split view of the secondary structure view and a linear sequence view.](image)

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 22.14).

![Figure 22.14: The blue circle represents the anchor point for rotating the substructure.](image)

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 22.15, the structure shown in figure 22.14 has been modified by dragging with the mouse.

![Figure 22.15: The structure has now been rotated.](image)

Press **Reset layout** in the **Side Panel** to reset the layout to the way it looked when the structure was predicted.

### 22.2.2 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 22.1), 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 22.2.1).

- 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 22.16.

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 22.5.2). Each substructure
Figure 22.16: 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 $\Delta G$. 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 $\Delta G$. 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 22.17).

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 22.18).

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 22.18 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 22.17: 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.
Figure 22.18: 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.

22.2.3 Symbolic representation in sequence view

In the Side Panel of normal sequence views ( ), you will find an extra group under Nucleotide info called Secondary Structure. This is used to display a symbolic representation of the secondary structure along the sequence (see figure 22.19).

Figure 22.19: 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 $\Delta G$.
  – 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 22.19).

• **Unpaired symbol.** How bases which are not part of a base pair should be represented (see figure 22.19).

• **Height.** When you zoom out, this option determines the height of the symbols as shown in figure 22.20 (when zoomed in, there is no need for specifying the height).

• **Base pair probability.** See section 22.2.4 below).

When you zoom in and out, the appearance of the symbols change. In figure 22.19, the view is zoomed in. In figure 22.20 you see the same sequence zoomed out to fit the width of the sequence.

![Figure 22.20: The secondary structure visualized below the sequence and with annotations shown above. The view is zoomed out to fit the width of the sequence.](image)

**22.2.4 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 22.4.

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.

### 22.3 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 22.1.2. By adding several annotations complex structural hypotheses can be formulated (see 22.21).

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:

\[
P(H) = \frac{\sum_{s_H \in S_H} P(s_H)}{\sum_{s \in S} P(s)} = \frac{PF_H}{PF_{full}},
\]

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. 22.21.

![Figure 22.21: Two constraints defining a structural hypothesis.](image)

### 22.3.1 Selecting sequences for evaluation

The evaluation is started from the Toolbox:

**Toolbox | RNA Structure | Evaluate Structure Hypothesis**

This opens the dialog shown in figure 22.22.

If you have selected sequences before choosing the Toolbox action, they are now listed in the Selected Elements window of the dialog. Use the arrows to add or remove sequences or 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 22.23).

The partition function algorithm includes a number of advanced options:

- **Avoid isolated base pairs.** The algorithm filters out isolated base pairs (i.e. stems of length 1).


- Apply different energy rules for Grossly Asymmetric Interior Loops (GAIL). Compute the minimum free energy applying different rules for Grossly Asymmetric Interior Loops (GAIL). A Grossly Asymmetric 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].

### 22.3.2 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 22.24).
22.4 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].

22.4.1 Selecting sequences for scanning

The scanning is started from the Toolbox:

Toolbox | RNA Structure ( ) | Evaluate Structure Hypothesis ( )

This opens the dialog shown in figure 22.25.

Figure 22.25: 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 22.26).

The first group of parameters pertain to the methods of sequence resampling. There are four ways of resampling, all described in detail in [Clote et al., 2005]:

- **Mononucleotide shuffling.** Shuffle method generating a sequence of the exact same mononucleotide frequency
- **Dinucleotide shuffling.** Shuffle method generating a sequence of the exact same dinucleotide frequency
- **Mononucleotide sampling from zero order Markov chain.** Resampling method generating a sequence of the same expected mononucleotide frequency.
• **Dinucleotide sampling from first order Markov chain.** Resampling method generating a sequence of the same expected dinucleotide frequency.

The second group of parameters pertain to the scanning settings and include:

• **Window size.** The width of the sliding window.

• **Number of samples.** The number of times the sequence is resampled to produce the background distribution.

• **Step increment.** Step increment when plotting sequence positions against scoring values.

The third parameter group contains the output options:

• **Z-scores.** Create a plot of Z-scores as a function of sequence position.

• **P-values.** Create a plot of the statistical significance of the structure signal as a function of sequence position.

**22.4.2 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 22.27).
Figure 22.27: A plot of the Z-scores produced by sliding a window along a sequence.

22.5 Bioinformatics explained: RNA structure prediction by minimum free energy minimization

RNA molecules are hugely important in the biology of the cell. Besides their rather simple role as an intermediate messenger between DNA and protein, RNA molecules can have a plethora of biologic functions. Well known examples of this are the infrastructural RNAs such as tRNAs, rRNAs and snRNAs, but the existence and functionality of several other groups of non-coding RNAs are currently being discovered. These include micro- (miRNA), small interfering- (siRNA), Piwi interacting- (piRNA) and small modulatory RNAs (smRNA) [Costa, 2007].

A common feature of many of these non-coding RNAs is that the molecular structure is important for the biological function of the molecule.

Ideally, biological function is best interpreted against a 3D structure of an RNA molecule. However, 3D structure determination of RNA molecules is time-consuming, expensive, and difficult [Shapiro et al., 2007] and there is therefore a great disparity between the number of known RNA sequences and the number of known RNA 3D structures.

However, as it is the case for proteins, RNA tertiary structures can be characterized by secondary structural elements. These are defined by hydrogen bonds within the molecule that form several recognizable "domains" of secondary structure like stems, hairpin loops, bulges and internal loops (see below). Furthermore, the high degree of base-pair conservation observed in the evolution of RNA molecules shows that a large part of the functional information is actually contained in the secondary structure of the RNA molecule.

Fortunately, RNA secondary structure can be computationally predicted from sequence data allowing researchers to map sequence information to functional information. The subject of this
paper is to describe a very popular way of doing this, namely free energy minimization. For an in-depth review of algorithmic details, we refer the reader to Mathews and Turner, 2006.

### 22.5.1 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 (\( S_{\text{min}} \)) 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 \( S_{\text{min}} \) 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 \( S_{\text{min}} \) 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 \( S_{\text{min}} \) - 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 22.28.

![Figure 22.28: 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.](image)

### 22.5.2 Structure elements and their energy contribution

In this section, we classify the structure elements defining a secondary structure and describe their energy contribution.
Figure 22.29: 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.

**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 elements:

1. **Hairpin loop** (G). 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 22.29A).

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 22.29B). 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 \( \geq 1 \) on the other side (see figure 22.29C). 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 \( \geq 1 \) on both sides (see figure 22.29D). 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 22.29E). 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 22.29F). 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 22.29G). 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 22.29H).

- **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 22.29I). 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 22.30) and it has been shown that this can dramatically increase the fidelity of the secondary structure prediction [Mathews and Turner, 2006].

![Diagram](CTCTTTAAACCATTTAATAAGTAAATTAGCAC)

Figure 22.30: *Known structural features can be added as constraints to the secondary structure prediction algorithm in CLC Main Workbench.*
Chapter 23

Expression analysis

Contents

23.1 Experimental design .................................................. 527
  23.1.1 Setting up a microarray experiment .......................... 527
  23.1.2 Organization of the experiment table ....................... 530
  23.1.3 Adding annotations to an experiment ....................... 535
  23.1.4 Scatter plot view of an experiment ......................... 536
  23.1.5 Cross-view selections ......................................... 538
23.2 Transformation and normalization ................................. 539
  23.2.1 Selecting transformed and normalized values for analysis .. 540
  23.2.2 Transformation ................................................. 540
  23.2.3 Normalization .................................................... 541
23.3 Quality control ....................................................... 543
  23.3.1 Create Box Plot ................................................ 543
  23.3.2 Hierarchical Clustering of Samples ......................... 547
  23.3.3 Principal Component Analysis ............................ 551
23.4 Feature clustering ................................................... 555
  23.4.1 Hierarchical clustering of features ....................... 555
  23.4.2 K-means/medoids clustering ............................... 560
23.5 Statistical analysis - identifying differential expression ........ 563
  23.5.1 Tests on proportions ......................................... 564
  23.5.2 Gaussian-based tests ....................................... 565
  23.5.3 Corrected p-values .......................................... 567
  23.5.4 Volcano plots - inspecting the result of the statistical analysis .. 568
23.6 Annotation tests .................................................... 570
  23.6.1 Hypergeometric Tests on Annotations ...................... 570
  23.6.2 Gene Set Enrichment Analysis ............................ 573
23.7 General plots ....................................................... 576
  23.7.1 Histogram ....................................................... 576
  23.7.2 MA plot .......................................................... 578
  23.7.3 Scatter plot .................................................. 581
CHAPTER 23. EXPRESSION ANALYSIS

This section focuses on analysing expression data from sources such as microarrays using tools found under the Expression Analysis (folder) of the Toolbox. This includes 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 also available to aid interpretation of the results.

Tools for analysing RNA-Seq and small RNA data are available in the CLC Genomics Workbench.

23.1 Experimental design

Central to expression analysis in the CLC Main Workbench is the concept of the sample. For microarray data, a sample would typically consist of the expression values from one array. Within a sample, there are a number of features, usually genes, and their associated expression levels.

Importing expression data into the Workbench as samples is described in appendix section I.

The first step towards analyzing this expression data is to create an Experiment, which contains information about which samples belong to which groups.

23.1.1 Setting up a microarray experiment

To analyze differential expression, the Workbench must know which samples belong to which groups. This information is provided in an Experiment. Statistical analyses can then be carried out using the information in the Experiment to investigate differential expression. The Experiment element is also where things like t-test results and clustering information are stored.

The statistics created using this tool is for microarray data, RNA-Seq differential expression are better handled by the tools in the folder Toolbox | Expression Analysis (folder).

To set up an experiment:

Toolbox | Expression Analysis (folder) | Set Up Microarray Experiment (button)

Select the samples that you wish to use by double-clicking or selecting and pressing the Add (button) (see figure 23.1).

Clicking Next shows the dialog in figure 23.2.

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.

  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]: 
\[
\text{RPKM} = \frac{\text{total exon reads}}{\text{mapped reads (millions) } \times \text{exon length (KB)}}.
\]
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 23.3.

Figure 23.3: **Naming the groups.**

Depending on the number of groups selected in figure 23.2, you will see a list of groups with text fields where you can enter an appropriate name for that group.

For multi-group experiments, if you find out that you have too many groups, click the **Delete** button. If you need more groups, simply click **Add New Group**.

Click **Next** when you have named the groups, and you will see figure 23.4.

This is where you define which group the individual sample belongs to. Simply select one or more samples (by clicking and dragging the mouse), right-click (Ctrl-click on Mac) and select the appropriate group.

Note that the samples are sorted alphabetically based on their names.

If you have chosen **Paired** in figure 23.2, 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.

### 23.1.2 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 23.5).

For a general introduction to table features like sorting and filtering, see section 3.2.

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 23.6).

![Figure 23.6: The initial view of the experiment level for a two-group experiment.](image)

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 23.2.3 and 23.2.2, respec-
tively), 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 23.5.4. There are different definitions
of 'Fold Change' in the literature. The definition that is used typically depends on the original
scale of the data that is analyzed. For data whose original scale is not the log scale the standard
definition is the ratio of the group means [Tusher et al., 2001]. This is the value you find in
the 'Fold Change' column of the experiment. However, for data whose original is the log scale,
the difference of the mean expression levels is sometimes referred to as the fold change [Guo
et al., 2006], and if you want to filter on fold change for these data you should filter on the
values in the 'Difference' column. Your data's original scale will e.g. be the log scale if you have
imported Affymetrix expression values which have been created by running the RMA algorithm on
the probe-intensities.
Analysis level

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 23.5).

An example of Analysis level settings is shown in figure 23.7.

![Figure 23.7: An example of columns available under the Analysis level section.](image)

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 23.1.3), the annotations will be listed in the Annotation level group as shown in figure 23.8.

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).

Group level

At the group level, you can show/hide entire groups (Heart and Diaphragm in figure 23.5). 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 23.2.3 and 23.2.2, respectively), the means of the normalized and transformed values will also appear.

An example is shown in figure 23.9.
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 23.2.3 and 23.2.2, respectively), the normalized and transformed values will also appear.

An example is shown in figure 23.10.
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 3.2).

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 23.11).

![Create Experiment from Selection button](image1)

Figure 23.11: 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 23.12).

![Download Sequence button](image2)

Figure 23.12: Select sequences and press the download button.

This will open a dialog where you specify where the sequences should be saved. You can learn more about opening and viewing sequences in chapter 12. You can now use the downloaded sequences for further analysis in the Workbench.

23.1.3 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 1. In order to associate an annotation file with an experiment, either select the annotation file when you set up the experiment (see section 23.1.1), 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 23.1.2. You can also add annotations by pressing the Add Annotations button at the bottom of the table (see figure 23.13).

![Figure 23.13: Adding annotations by clicking the button at the bottom of the experiment table.](image)

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 23.14.

![Figure 23.14: Choosing how to match annotations with samples.](image)

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.

### 23.1.4 Scatter plot view of an experiment

At the bottom of the experiment table, you can switch between different views of the experiment (see figure 23.15).

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 23.16.

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.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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 or Wide
• **Line type** None, Line, Long dash or Short dash
• **Line color** 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** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.
• **Dot color** 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).

### 23.1.5 Cross-view selections

There are a number of different ways of looking at an experiment as shown in figure 23.17.

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 23.17, 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 23.18.

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 heat map, and all other open views will reflect the selection.

A common use of the split views is where you have an experiment and have performed a statistical analysis. You filter the experiment to identify all genes that have an FDR corrected p-value below 0.05 and a fold change for the test above say, 2. You can select all the rows in the experiment table satisfying these filters by holding down the Cntrl button and clicking 'a'. If you have a split view of the experiment and the volcano plot all points in the volcano plot corresponding to the selected features will be red. Note that the volcano plot allows two sets of values in the columns...
under the test you are considering to be displayed on the x-axis: the 'Fold change’s and the 'Difference’s. You control which to plot in the side panel. If you have filtered on 'Fold change’ you will typically want to choose 'Fold change’ in the side panel. If you have filtered on 'Difference’ (e.g. because your original data is on the log scale, see the note on fold change in 23.1.2) you typically want to choose 'Difference’.

23.2 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).
23.2.1 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 23.19).

![Figure 23.19: 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.](image)

In this case, the values have not been normalized, so it is not possible to select normalized values.

23.2.2 Transformation

The CLC Main Workbench lets you transform expression values based on logarithm and adding a constant:

![Toolbox | Expression Analysis | Transformation and Normalization | Transform](image)

Select a number of samples or an experiment and click Next.

This will display a dialog as shown in figure 23.20.

![Figure 23.20: Transforming expression values.](image)

At the top, you can select which values to transform (see section 23.2.1).

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.

### 23.2.3 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 **23.21**.

![Normalization dialog](image)

**Figure 23.21:** 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 23.22 and 23.23 show the effect on the distribution of expression values when using scaling or quantile normalization, respectively.

![Figure 23.22: Box plot after scaling normalization.](image)

![Figure 23.23: Box plot after quantile normalization.](image)

At the bottom of the dialog in figure 23.21, you can select which values to normalize (see section 23.2.1).

Clicking **Next** will display a dialog as shown in figure 23.24.

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.
23.3 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.

23.3.1 Create Box Plot

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:

Click Finish to start the tool.
Select a number of samples ( ) or ( ) or an experiment ( ) and click **Next**. This will display a dialog as shown in figure 23.25.

![Figure 23.25: Choosing values to analyze for the box plot.](image)

Here you select which values to use in the box plot (see section 23.2.1). 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 23.26.

![Figure 23.26: A box plot of 12 samples in a two-group experiment, colored by group.](image)

Note that the boxes 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. The box also includes the IQR values (from the lower to the upper quartile) and the median is displayed as a line in the box. The ends of the boxplot whiskers are the lowest data point within 1.5 times the inter quartile range (IQR) of the lower quartile and the highest data point within 1.5 IQR of the upper quartile. It is possible to change the default value of 1.5 using the side panel option "Whiskers range factor".

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 23.27).
**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.

**Tick lines at** Choosing Major ticks will show a grid behind the graph.

**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 23.28).
• **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** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.

• **Dot color** 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 box plots**

This section will show how to interpret a box plot through a few examples.

First, if you look at figure 23.29, you can see a box plot for an experiment with 5 groups and 27 samples.

![Box Plot](image)

*Figure 23.29: 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 23.30 shows a box plot for the same experiment after quantile normalization: the distributions have been brought into par.

In figure 23.31 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.
23.3.2 Hierarchical Clustering of Samples

A hierarchical clustering of samples is a tree representation of their relative similarity. The tree structure is generated by

1. letting each 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 23.32. 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 23.32: Parameters for hierarchical clustering of samples.](image)

There are three kinds of **Distance measures**:

- **Euclidean distance.** The ordinary distance between two points - the length of the segment connecting them. If \( u = (u_1, u_2, \ldots, u_n) \) and \( v = (v_1, v_2, \ldots, v_n) \), then the Euclidean distance between \( u \) and \( v \) is

  \[
  |u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.
  \]

- **1 - Pearson correlation.** The Pearson correlation coefficient between two elements \( x = (x_1, x_2, \ldots, x_n) \) and \( y = (y_1, y_2, \ldots, y_n) \) is defined as

  \[
  r = \frac{1}{n-1} \sum_{i=1}^{n} \left( \frac{x_i - \bar{x}}{s_x} \right) \times \left( \frac{y_i - \bar{y}}{s_y} \right)
  \]

  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 \( 1 - |Pearson correlation| \) as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance.** The Manhattan distance between two points is the distance measured along axes at right angles. If \( u = (u_1, u_2, \ldots, u_n) \) and \( v = (v_1, v_2, \ldots, v_n) \), then the Manhattan distance between \( u \) and \( v \) is

  \[
  |u - v| = \sum_{i=1}^{n} |u_i - v_i|.
  \]
The possible cluster linkages 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 23.2.1).

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 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 23.33.

![Sample clustering](image)

**Figure 23.33: 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 23.34).

![Show Heat Map](image)

**Figure 23.34: Showing the hierarchical clustering of an experiment.**
If you have run the workflow version of the tool, or selected a number of samples 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 23.33, 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 23.4.1).

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 unclean tissue composition or samples for which the processing has gone wrong. Unexpectedly placed samples, of course, could also be highly interesting samples.

There are a number of options to change the appearance of the heat map. At the top of the Side Panel, you find the Heat map preference group (see figure 23.35).

![Figure 23.35: Side Panel of heat map.](image)

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 23.36).

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 heat map. 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.

For clustering of features, the **Features** group has an option to "Optimize tree layout". This attempts to reorder the features, consistently with the tree, such that the most expressed features form a diagonal from the top-left to the bottom-right of the heat map.

The **Samples** group contains an "Order by:" dropdown that allows re-ordering of the columns of the heat map. When clustering by samples it is possible to choose between using the "Tree" to determine the sample ordering, and showing the "Samples" in the order they were input to the tool. When clustering by features, only the "Samples" input order is available.

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).

### 23.3.3 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 ( ) or an experiment ( ) and click Next.

This will display a dialog as shown in figure 23.37.

![Figure 23.37: Selecting which values the principal component analysis should be based on.](image)

In this dialog, you select the values to be used for the principal component analysis (see
section 23.2.1).

Click Finish to start the tool.

Principal component analysis plot

This will create a principal component plot as shown in figure 23.38.

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
Figure 23.38: A principal component analysis.

that uses 'log-scale' data).

The plot in figure 23.38 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.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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 or Wide
- **Line type** None, Line, Long dash or Short dash
- **Line color** Click the color box to select a color.

Below the general preferences, you find the **Dot properties**:

- **Select sample or group**. When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.

- **Dot type** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.

- **Dot color** 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 23.38, 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.

- **Tick lines at** Choosing Major ticks will show a grid behind the graph.

- **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** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.
- **Dot color** Click the color box to select a color.
- **Line width** Thin, Medium or Wide
- **Line type** None, Line, Long dash or Short dash
- **Line color** 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).

### 23.4 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.

#### 23.4.1 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 23.1.2.

Clicking **Next** will display a dialog as shown in figure 23.39. The hierarchical clustering algorithm requires that you specify a distance measure and a cluster linkage. The distance measure is used to 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.

There are three kinds of **Distance measures:**

- **Euclidean distance.** The ordinary distance between two points - the length of the segment connecting them. If \( u = (u_1, u_2, \ldots, u_n) \) and \( v = (v_1, v_2, \ldots, v_n) \), then the Euclidean distance between \( u \) and \( v \) is

\[
|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.
\]

- **1 - Pearson correlation.** The Pearson correlation coefficient between two elements \( x = (x_1, x_2, \ldots, x_n) \) and \( y = (y_1, y_2, \ldots, y_n) \) is defined as

\[
r = \frac{1}{n - 1} \sum_{i=1}^{n} \left( \frac{x_i - \bar{x}}{s_x} \right) \cdot \left( \frac{y_i - \bar{y}}{s_y} \right)
\]

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 \( 1 - |Pearson\ correlation| \) as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.
• **Manhattan distance.** The Manhattan distance between two points is the distance measured along axes at right angles. If \( u = (u_1, u_2, \ldots, u_n) \) and \( v = (v_1, v_2, \ldots, v_n) \), then the Manhattan distance between \( u \) and \( v \) is
\[
|u - v| = \sum_{i=1}^{n} |u_i - v_i|.
\]

The possible cluster linkages 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 23.2.1). Click **Finish** to start the tool.

**Result of hierarchical clustering of features**

The result of a feature clustering is shown in figure 23.40.

![Hierarchical clustering of features](image)

**Figure 23.40: Hierarchical clustering of features.**

If you have used an experiment as input, the clustering is added to the experiment and will be saved when you save the experiment. It can be viewed by clicking the **Show Heat Map** button at the bottom of the view (see figure 23.41).
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 23.40). 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 23.42).

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 23.43).

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.
Figure 23.43: When more than one clustering has been performed, there will be a list of heat maps to choose from.

- **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 heat map. 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.

For clustering of features, the Features group has an option to "Optimize tree layout". This attempts to reorder the features, consistently with the tree, such that the most expressed features form a diagonal from the top-left to the bottom-right of the heat map.

The Samples group contains an "Order by:" dropdown that allows re-ordering of the columns of the heat map. When clustering by samples it is possible to choose between using the "Tree" to determine the sample ordering, and showing the "Samples" in the order they were input to the tool. When clustering by features, only the "Samples" input order is available.

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).
23.4.2 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 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 23.1.2.

Clicking Next will display a dialog as shown in figure 23.44.

![Figure 23.44: Parameters for k-means/medoids clustering.](image)

The parameters are:

- **Algorithm.** You can choose between two clustering methods:
  - **K-means.** K-means clustering assigns each point to the cluster whose center is nearest. The center/centroid of a cluster is defined as the average of all points in the cluster. If a data set has three dimensions and the cluster has two points $X = (x_1, x_2, x_3)$ and $Y = (y_1, y_2, y_3)$, then the centroid $Z$ becomes $Z = (z_1, z_2, z_3)$, where $z_i = (x_i + y_i) / 2$ for $i = 1, 2, 3$. The algorithm attempts to minimize the intra-cluster variance defined by:

$$ V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 $$

where there are $k$ clusters $S_i, i = 1, 2, \ldots, k$ and $\mu_i$ is the centroid of all points $x_j \in S_i$. The detailed algorithm can be found in [Lloyd, 1982].
K-medoids. K-medoids clustering is computed using the PAM-algorithm (PAM is short for Partitioning Around Medoids). It chooses datapoints as centers in contrast to the K-means algorithm. The PAM-algorithm is based on the search for $k$ representatives (called medoids) among all elements of the dataset. When having found $k$ representatives $k$ clusters are now generated by assigning each element to its nearest medoid. The algorithm first looks for a good initial set of medoids (the BUILD phase). Then it finds a local minimum for the objective function:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - c_i)^2$$

where there are $k$ clusters $S_i$, $i = 1, 2, \ldots, k$ and $c_i$ is the medoid of $S_i$. This solution implies that there is no single switch of an object with a medoid that will decrease the objective (this is called the SWAP phase). The PAM-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 $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean distance between $u$ and $v$ is
    $$|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.$$
  - **Manhattan distance.** The Manhattan distance between two elements is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between $u$ and $v$ is
    $$|u - v| = \sum_{i=1}^{n} |u_i - v_i|.$$
- **Subtract mean value.** For each gene, subtract the mean gene expression value over all input samples.

Clicking **Next** will display a dialog as shown in figure 23.45.

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 23.2.1).

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 23.44) - there is one graph per cluster. Using drag and drop as explained in section 2.1.4, you can arrange the views to see more than one graph at the time.

Figure 23.46 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 23.1.5.
23.5 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 23.5.1) and tests that assume that the data is real-valued, has Gaussian distributions and compare means (described in section 23.5.2).

To run the statistical analysis:

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 23.1.1).
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 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.

23.5.1 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 23.1.1), 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.

23.5.2 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 23.47.

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 23.1.1) 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 23.1.1) 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.

### 23.5.3 Corrected p-values

Clicking **Next** will display a dialog as shown in figure 23.48.

![Figure 23.48: Additional settings for the statistical analysis.](image)

At the top, you can select which values to analyze (see section 23.2.1).

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.

### 23.5.4 Volcano plots - inspecting the result of the statistical analysis

The results of the statistical analysis are added to the experiment and can be shown in the experiment table (see section 23.1.2). 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 3.2).

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 23.1.4, 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 23.49.

The volcano plot shows the relationship between the p-values of a statistical test and the magnitude of the difference in expression values of the samples in the groups. On the y-axis the $-\log_{10}$ p-values are plotted. For the x-axis you may choose between two sets of values by choosing either 'Fold change' or 'Difference' in the volcano plot side panel’s 'Values' part. If you choose 'Fold change' the log of the values in the 'fold change' (or 'Weighted fold change') column for the test will be displayed. If you choose 'Difference' the values in the 'Difference' (or 'Weighted difference') column will be used. Which values you wish to display will depend upon the scale of your data (Read the note on fold change in section 23.1.2).

The larger the difference in expression of a feature, the more extreme it’s point will lie on the X-axis. The more significant the difference, the smaller the p-value and thus the higher
the $-\log_{10}(p)$ value. Thus, points for features with highly significant differences will lie high in the plot. Features of interest are typically those which change significantly and by a certain magnitude. These are the points in the upper left and upper right hand parts of the volcano plot.

If you have performed different tests or you have an experiment with multiple groups you need to specify for which test and which group comparison you want the volcano plot to be shown. You do this in the ‘Test’ and ‘Values’ parts of the volcano plot side panel.

Options for the volcano plot are described in further detail when describing the Side Panel below.

If you place your mouse on one of the dots, a small text box will tell the name of the feature. Note that you can zoom in and out on the plot (see section 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.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.
- **Dot color** 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 23.1.2.

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).

### 23.6 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 23.1.3.

#### 23.6.1 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].
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 23.1.2).

Click Next. This will display the dialog shown in figure 23.50.

Figure 23.50: 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 23.2.1). 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.

Click **Finish** to start the tool.

The final number of features used for the test is reported in this history view of the test results.

**Result of hypergeometric tests on annotations** The result of performing hypergeometric tests
on annotations using GO biological process is shown in figure 23.51.

The table shows the following information:

- **Category**. This is the identifier for the category.
- **Description**. This is the description belonging to the category. Both of these are simply
  extracted from the annotations.
- **Full set**. The number of features in the original experiment (not the subset) with this
  category. (Note that this is after removal of duplicates).
- **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'

Figure 23.51: The result of testing on GO biological process.
• **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.

Note that when testing for the significance of a particular GO term, we take into account that GO has a hierarchical structure. See section ?? for a detailed description on how to interpret potential discrepancies in the number of features in your results and the original GAF file.

### 23.6.2 Gene Set Enrichment Analysis

When carrying out a hypergeometric test on annotations you typically compare the annotations of the genes in a subset containing ‘the significantly differentially expressed genes’ to those of the total set of genes in the experiment. Which, and how many, genes are included in the subset is somewhat arbitrary - using a larger or smaller p-value cut-off will result in including more or less. Also, the magnitudes of differential expression of the genes is not considered.

The Gene Set Enrichment Analysis (GSEA) does NOT take a sublist of differentially expressed genes and compare it to the full list - it takes a single gene list (a single experiment). The idea behind GSEA is to consider a measure of association between the genes and phenotype of interest (e.g. test statistic for differential expression) and rank the genes according to this measure of association. A test is then carried out for each annotation category, for whether the ranks of the genes in the category are evenly spread throughout the ranked list, or tend to occur at the top or bottom of the list.

The GSEA test implemented here is that of [Tian et al., 2005]. The test implicitly calculates and uses a standard t-test statistic for two-group experiments, and ANOVA statistic for multiple group experiments for each feature, as measures of association. For each category, the test statistics for the features in than category are summed and a category based test statistic is calculated as this sum divided by the square root of the number of features in the category. Note that if a feature has the value NaN in one of the samples, the t-test statistic for the feature will be NaN. Consequently, the combined statistic for each of the categories in which the feature is included will be NaN. Thus, it is advisable to filter out any feature that has a NaN value before applying GSEA.

The p-values for the GSEA test statistics are calculated by permutation: The original test statistics for the features are permuted and new test statistics are calculated for each category, based on the permuted feature test statistics. This is done the number of times specified by the user in the wizard. For each category, the lower and upper tail probabilities are calculated by comparing the original category test statistics to the distribution of the permutation-based test statistics for that category. The lower and higher tail probabilities are the number of these that are lower and higher, respectively, than the observed value, divided by the number of permutations.

As the p-values are based on permutations you may some times see results where category x’s test statistic is lower than that of category y and the categories are of equal size, but where the lower tail probability of category x is higher than that of category y. This is due to imprecision in the estimations of the tail probabilities from the permutations. The higher the number of permutations, the more stable the estimation.

You may run a GSEA on a full experiment, or on a sub-experiment where you have filtered away...
features that you think are un-informative and represent only noise. Typically you will remove features that are constant across samples (those for which the value in the ‘Range’ column is zero’ — these will have a t-test statistic of zero) and/or those for which the inter-quantile range is small. As the GSEA algorithm calculates and ranks genes on p-values from a test of differential expression, it will generally not make sense to filter the experiment on p-values produced in an analysis if differential expression, prior to running GSEA on it.

Toolbox | Expression Analysis () | Annotation Test () | Gene Set Enrichment Analysis (GSEA) ()

Select an experiment and click Next.

Click Next. This will display the dialog shown in figure 23.52.

Figure 23.52: 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 23.53.

![Figure 23.53: Gene set enrichment analysis parameters.](image)

At the top, you can select which values to analyze (see section 23.2.1).

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 23.54.

![Figure 23.54: The result of gene set enrichment analysis on GO biological process.](image)

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. Note that when testing for the significance of a particular GO term, we take into account that GO has a hierarchical structure. See section ?? for a detailed description on how to interpret potential discrepancies in the number of genes in your results and the original GAF file.

### 23.7 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.

#### 23.7.1 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 (Expression Analysis)](Toolbox | Expression Analysis (Expression Analysis)) | [General Plots](General Plots) | [Create Histogram](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 **23.55**.

![Figure 23.55: Selecting which values the histogram should be based on.](Figure 23.55: 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 **23.2.1**). Click **Finish** to start the tool.
Viewing histograms

The resulting histogram is shown in a figure 23.56

Figure 23.56: Histogram showing the distribution of transformed expression values.

The histogram shows the expression value on the x axis (in the case of figure 23.56 the transformed expression values) and the counts of these values on the y axis.

In the Side Panel to the left, there is a number of options to adjust the view. Under Graph preferences, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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 23.57.

![Figure 23.57: Table view of a histogram.](image)

The table lists the following properties:

- **Number +Inf values**
- **Number -Inf values**
- **Number NaN values**
- **Number values used**
- **Total number of values**

### 23.7.2 MA plot

The MA plot is a scatter rotated by $45^\circ$. 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 23.58.

In this dialog, you select the values to be used for creating the MA plot (see section 23.2.1). Click **Finish** to start the tool.
CHAPTER 23. EXPRESSION ANALYSIS

Viewing MA plots

The resulting plot is shown in a figure 23.59.

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 23.59 it is clear that the variance increases with the mean. With an MA plot like this, you will often choose to transform the expression values (see section 23.2.2).

Figure 23.60 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.

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.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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 or Wide
  - **Line type** None, Line, Long dash or Short dash
  - **Line color** Click the color box to select a color.

- **Line width** Thin, Medium or Wide
- **Line type** None, Line, Long dash or Short dash
- **Line color** 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** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.
- **Dot color** 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).
23.7.3 Scatter plot

As described in section 23.1.4, 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 23.61.

Figure 23.61: 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 23.2.1). Click Finish to start the tool.

For more information about the scatter plot view and how to interpret it, please see section 23.1.4.
# Chapter 24

## BLAST search

### Contents

- **24.1 Running BLAST searches** ........................................ 583
  - 24.1.1 BLAST at NCBI ........................................ 583
  - 24.1.2 BLAST against local data ................................ 586
- **24.2 Output from BLAST searches** ............................ 590
  - 24.2.1 Graphical overview for each query sequence .......... 590
  - 24.2.2 Overview BLAST table .................................. 590
  - 24.2.3 BLAST graphics ......................................... 592
  - 24.2.4 BLAST HSP table ...................................... 593
  - 24.2.5 BLAST hit table ........................................ 595
  - 24.2.6 Extracting a consensus sequence from a BLAST result 596
- **24.3 Local BLAST databases** ................................. 596
  - 24.3.1 Make pre-formatted BLAST databases available ........ 596
  - 24.3.2 Download NCBI pre-formatted BLAST databases .... 597
  - 24.3.3 Create local BLAST databases ........................ 598
- **24.4 Manage BLAST databases** .............................. 599
- **24.5 Bioinformatics explained: BLAST** ...................... 600
  - 24.5.1 How does BLAST work? ................................ 601
  - 24.5.2 Which BLAST program should I use? .................... 603
  - 24.5.3 Which BLAST options should I change? .............. 604
  - 24.5.4 Where can I get the BLAST+ programs ........... 605
  - 24.5.5 What you cannot get out of BLAST ................. 605
  - 24.5.6 Other useful resources .............................. 606

**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 24.5.

Figure 24.8 shows an example of a BLAST result in the CLC Main Workbench.

**24.1 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.

**24.1.1 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 24.2

Select one or more sequences of the same type (either DNA or protein) and click **Next**.
In this dialog, you choose which type of BLAST search to conduct, and which database to search against (figure 24.3). The databases at the NCBI listed in the dropdown box will correspond to the query sequence type you have, DNA or protein, and the type of blast search you can chose among to run. A complete list of these databases can be found in Appendix B. Here you can also read how to add additional databases available the NCBI to the list provided in the dropdown menu.

**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 24.4, allows you to choose parameters to tune your BLAST search, to meet your requirements.

![Figure 24.4: Parameters that can be set before submitting a BLAST search.](image)

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.


- **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](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.

- **Mask low complexity regions.** Mask off segments of the query sequence that have low compositional complexity.

- **Expect.** The threshold for reporting matches against database sequences. The Expect value (E-value) describes the number of hits one can expect to see matching a query by
chance when searching against a database of a given size. If the E-value ascribed to a match is greater than the value entered in the Expect field, the match will not be reported. Details of how E-values are calculated can be found at the NCBI: [http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html](http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html). Lower 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 lower than 1 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 24.3 and the rest of the options are the same as when performing a BLAST search with a full sequence.

**24.1.2 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 24.3.3)
• You can add pre-formatted databases (see section 24.3.1)
• 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 24.5:

![Figure 24.5: Choose one or more sequences to conduct a BLAST search.](image)

Select one or more sequences of the same type (DNA or protein) and click Next.

This opens the dialog seen in figure 24.6:

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.
• **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 13.1.3).

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 24.3.3.

• **BLAST Database.** Select a database already available in one of your designated BLAST database folders. Read more in section 24.4.

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 24.7).
Figure 24.7: 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.

- **Mask low complexity regions.** 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 Expect value (E-value) describes the number of hits one can expect to see matching a query by chance when searching against a database of a given size. If the E-value ascribed to a match is greater than the value entered in the Expect field, the match will not be reported. Details of how E-values are calculated can be found at the NCBI: [http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html](http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html). Lower 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 lower than 1 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.

- **Filter out redundant results.** This option culls HSPs on a per subject sequence basis by removing HSPs that are completely enveloped by another HSP.

**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 24.6 and the rest of the options are the same as when performing a BLAST search with a full sequence.

### 24.2 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 24.2.2.

#### 24.2.1 Graphical overview for each query sequence

Double clicking on a given row of a tabular blast table opens a graphical overview of the blast results for a particular query sequence, as shown in figure figure 24.8. In cases where only one sequence was entered into a BLAST search, such a graphical overview is the default output.

Figure 24.8 shows an example of a BLAST result for an individual query sequence in the CLC Main Workbench.

Detailed descriptions of the overview BLAST table and the graphical BLAST results view are described below.

#### 24.2.2 Overview BLAST table

In the overview BLAST table for a multi-sequence blast search, as shown in figure 24.9, there is one row for each query sequence. Each row represents the BLAST result for this query sequence.

Double-clicking a row will open the BLAST result for this query sequence, allowing more detailed investigation of the result. You can also select one or more rows and click the **Open BLAST Output** button at the bottom of the view. Consensus sequence can be extracted by clicking the **Extract Consensus** button at the bottom. Clicking the **Open Query Sequence** will open a
Figure 24.8: 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.

Figure 24.9: An overview BLAST table summarizing the results for a number of query sequences. 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 %)
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.

### 24.2.3 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 12.2.

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.

### 24.2.4 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 24.10 is an example of a BLAST HSP Table.

![Figure 24.10: BLAST HSP Table](image)

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. Note that the 3D view has special system requirements, see section 1.3.

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 24.2.3 so that selecting a HSP in the table will make a selection on the corresponding sequence in the graphical view.

### 24.2.5 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 24.11 is an example of a BLAST Hit Table.

![Figure 24.11: BLAST Hit Table. The hits can be sorted by the different columns, simply by clicking the column heading.](image)

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.

### 24.2.6 Extracting a consensus sequence from a BLAST result

A consensus sequence can be extracted from nucleotide BLAST results, as described in section 19.10. That section focuses on working with read mappings, but the same underlying tool is used to extract consensus sequences from nucleotide BLAST results.

### 24.3 Local BLAST databases

BLAST databases on your local system can be made available for searches via your CLC Main Workbench (see section 24.3.1). To make adding databases even easier, you can download pre-formatted BLAST databases from the NCBI from within your Workbench (section 24.3.2). You can also easily create your own local blast databases from sequences within your Workbench (section 24.3.3).

#### 24.3.1 Make pre-formatted BLAST databases available

To use databases that have been downloaded or created outside the Workbench, you can either:

- Put the database files in one of the locations defined in the BLAST database manager (see section 24.4). 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 24.12).
- Add the location where your BLAST databases are stored using the BLAST database manager (see section 24.4).
CHAPTER 24. BLAST SEARCH

24.3.2 Download NCBI pre-formatted BLAST databases

Many popular pre-formatted databases are available for download from the NCBI. You can download any of the databases available from the list at ftp://ftp.ncbi.nlm.nih.gov/blast/db/ from within your 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 24.13 pops up showing you the list of databases 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 24.4).
There are two very important things to note if you wish to take advantage of this tool.

- Many of the databases listed are very large. Please make sure you have space for them. If you are working on a shared system, we recommend you discuss your plans with your system administrator and fellow users.

- Some of the databases listed are dependent on others. This will be listed in the Dependencies column of the Download BLAST Databases window. This means that while the database your are interested in may seem very small, it may require that you also download a very big database on which it depends.

An example of the second item above is Swissprot. To download a database from the NCBI that would allow you to search just Swissprot entries, you need to download the whole nr database in addition to the entry for Swissprot.

### 24.3.3 Create local BLAST databases

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 24.1.2).

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 24.3.1.

To create a BLAST database, go to:

**Toolbox | BLAST | Create BLAST Database**

This opens the dialog seen in figure 24.14.

![Create BLAST Database](image)

**Figure 24.14: 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 24.15, 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.** A short description. This is displayed along with the database name in the list of available databases when launching a local BLAST search. If no description is entered, the creation date is used as the description.
• **Location.** 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 24.4.

![Create BLAST Database](image)

Figure 24.15: **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 24.4, and when running local BLAST (see section 24.1.2).

**Create BLAST Database** creates BLAST+ version 4 (dbV4) databases.

**Sequence identifiers and BLAST databases**

Restrictions on sequence identifier lengths, format, and duplicates present in the underlying BLAST+ program for making databases, `makeblastdb`, do not apply when making databases using **Create BLAST Database**.

Internal handling of sequence names, introduced in version 21.0, allows this level of naming flexibility with newer versions of BLAST+. This, however, has the implication that databases created using **Create BLAST Databases** in CLC Main Workbench, CLC Genomics Workbench or CLC Genomics Server version 21.0 and later are intended for use only with the BLAST search tools in these software versions.

There should be no obvious effects of this internal handling of sequence names on local BLAST search results, including the names written to BLAST reports.

### 24.4 Manage BLAST databases

The BLAST databases available as targets for running local BLAST searches (see section 24.1.2) can be managed through the Manage BLAST Databases dialog (see figure 24.16):

![Manage BLAST Databases](image)

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 24.3.3, or they can be pre-formatted BLAST databases.

The list of locations can be modified using the **Add Location** and **Remove Location** buttons. Once the Workbench has scanned the locations, it will keep a cache of the databases (in order to improve performance). If you have added new databases that are not listed, you can press **Refresh Locations** to clear the cache and search the database locations again.
CHAPTER 24. BLAST SEARCH

Figure 24.16: Overview of available BLAST databases.

Note: The BLAST database location and all folders in its path should not have any spaces in their names.

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.

24.5 Bioinformatics explained: BLAST

BLAST (Basic Local Alignment Search Tool) has become the defacto standard in search and alignment tools [Altschul et al., 1990]. The BLAST algorithm is still actively being developed and is one of the most cited papers ever written in this field of biology. Many researchers use BLAST as an initial screening of their sequence data from the laboratory and to get an idea of what they are working on. BLAST is far from being basic as the name indicates; it is a highly advanced algorithm which has become very popular due to availability, speed, and accuracy. In short,
BLAST search programs look for potentially homologous sequences to your query sequences in databases, either locally held databases or those hosted elsewhere, such as at the NCBI (http://www.ncbi.nlm.nih.gov/) [McGinnis and Madden, 2004].

BLAST can be used for a lot of different purposes. Some of the most popular purposes are listed on the BLAST webpage at the NCBI: https://blast.ncbi.nlm.nih.gov/Blast.cgi.

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.

A BLAST search generates a report specifying the potentially homologous sequences found and their local alignments with the query sequence.

24.5.1 How does BLAST work?

BLAST identifies homologous sequences using a heuristic method which initially finds short matches between two sequences. After finding initial matches, BLAST attempts to build local alignments with the query sequence using these. Thus, BLAST does not guarantee the optimal alignment and some sequence hits may be missed. To find optimal alignments, the Smith-Waterman algorithm should be used (see below). Below, 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 24.17 for an illustration of words in a protein sequence.

Figure 24.17: 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 24.17). 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 24.18). 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 24.18: Neighborhood BLAST words based on the BLOSUM62 matrix. Only words where the threshold $T$ exceeds 13 are included in the initial seeding.

After the initial finding of words (seeding), the BLAST algorithm will extend the (only 3 residues long) alignment in both directions (see figure 24.19). Each time the alignment is extended, an alignment score is increased/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 24.19: 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 24.20.

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.
Figure 24.20: Each dot represents a word match. Increasing the threshold of T limits the search space significantly.

### 24.5.2 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, tblastx included in CLC software:

<table>
<thead>
<tr>
<th>Option</th>
<th>Query Type</th>
<th>DB Type</th>
<th>Comparison</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>blastn</td>
<td>Nucleotide</td>
<td>Nucleotide</td>
<td>Nucleotide-Nucleotide</td>
<td></td>
</tr>
<tr>
<td>blastp</td>
<td>Protein</td>
<td>Protein</td>
<td>Protein-Protein</td>
<td></td>
</tr>
<tr>
<td>tblastn</td>
<td>Protein</td>
<td>Nucleotide</td>
<td>Protein-Protein</td>
<td>The database is translated into protein</td>
</tr>
<tr>
<td>blastx</td>
<td>Nucleotide</td>
<td>Protein</td>
<td>Protein-Protein</td>
<td>The queries are translated into protein</td>
</tr>
<tr>
<td>tblastx</td>
<td>Nucleotide</td>
<td>Nucleotide</td>
<td>Protein-Protein</td>
<td>The queries and database are translated into protein</td>
</tr>
</tbody>
</table>

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.
24.5.3 Which BLAST options should I change?

There are a number of options that can be configured when using BLAST search programs. Setting these options to relevant values can have a great impact on the search result. A few of the key settings are described briefly below.

The E-value  The expect value (E-value) describes the number of hits one can expect to see matching the query by chance when searching against a database of a given size. An E-value of 1 can be interpreted as meaning that in a search like the one just run, you could expect to see 1 match of the same score by chance once. That is, a match that is not homologous to the query sequence. When looking for very similar sequences in a database, it is often beneficial to use very low E-values.

E-values depend 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. Below are some comments on what one could infer from results with E-values in particular ranges.

- **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 matches the hit sequence.
- **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 homolog, but it is a gray area.
- **E-value > 1** Proteins are most likely not related
- **E-value > 10** Hits are most likely not related 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 a 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 ‘fftffllllss’, 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  Changing 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.

The BLAST search programs at the NCBI adjust settings automatically when short sequences are being used for searches, and there is a dedicated page, Primer-BLAST, for searching for primer sequences. https://blast.ncbi.nlm.nih.gov/Blast.cgi.

**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.

### 24.5.4 Where can I get the BLAST+ programs

The BLAST+ package can be downloaded for use on your own computer, institution computer cluster or similar from ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/. It is available for a wide range of different operating systems.


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 24.21). It is also much easier to batch download a selection of hit sequences for further inspection.

![Figure 24.21: 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.](image)

### 24.5.5 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.

24.5.6 Other useful resources

The NCBI BLAST web page

The latest BLAST+ release

Download pages for pre-formatted BLAST databases

O’Reilly book on BLAST
http://www.oreilly.com/catalog/blast/
Chapter 25

Utility tools

Contents

25.1 Extract Annotated Regions ................................................. 607
25.2 Create Sequence List .................................................. 609
25.3 Update Sequence Attributes in Lists ............................... 609
25.4 Split Sequence List ...................................................... 613
25.5 Rename Elements ....................................................... 615
25.6 Rename Sequences in Lists ........................................... 619

25.1 Extract Annotated Regions

Using Extract Annotated Regions, parts of a sequence (or several sequences) can be extracted based on annotations. Lengths of flanking regions can be specified if desired. The output is a sequence list that contains sequences carrying the annotation specified.

Some examples of the use of this tool:

- Extract all tRNA gene sequences from a genome.
- Extract sequences for regions where annotations contain particular text

To launch Extract Annotated Regions, go to:

Toolbox | Utility Tools ( ) | Extract Annotated Regions ( )

This opens the wizard. In the first step (figure 25.1) you can select one or more sequences to extract annotated regions from.

Note: With the CLC Main Workbench, annotated sequences are expected as input. Options relating to track-based input are intended for the CLC Genomics Workbench, where this tool is also present.

At the top of the next dialog step (figure 25.2) you can specify which annotations to use.
• **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 where both "MLH1" and "Human" are included.

• **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.

### 25.2 Create Sequence List

For information on creating sequence lists, please see section 12.1.

### 25.3 Update Sequence Attributes in Lists

**Update Sequence Attributes in Lists** updates information pertinent to sequences within a sequence list. For example, descriptions can be updated, or new information types can be added. The attribute information to add to the sequences within a sequence list is provided via an Excel file, a comma separated text file, or a tab separated text file. The attribute/column to use to match upon, so that information is the relevant row is added to a particular sequence, is specified when launching the tool.

Attribute information for each sequence can be seen when viewing a sequence list in table view. Many attributes can be directly edited, or updated using the **Update Sequence Attributes in Lists** tool. Some, however, cannot be, for example the length or the start of the sequence, as these are characteristics of the sequence itself.

Individual values can be updated manually by right-clicking in the relevant cell and choosing to edit that attribute (figure 25.3). Working with editable attributes in tables is described in section 3.2.

Alternatively, right click on an individual sequence in the sequence list and choose to open that sequence. Then navigate to the Element info view and change attribute values there. Changes made in the Element info view are reflected immediately in the open sequence list.

For updating information for many sequences, the **Update Sequence Attributes in Lists** is recommended.

To launch the **Update Sequence Attributes in Lists** tool, go to:

**Toolbox | Utility Tools ( ) | Sequence Lists ( ) | Update Sequence Attributes in Lists ( )**

and select a sequence list as input (figure 25.4).

Multiple sequence lists of the same type (nucleotide or peptide) can be selected as input, however please note that sequences in all lists are considered together as a single input, and a single sequence list will be output, containing all sequences of the input lists.
Figure 25.3: Attributes on individual sequences in a sequence list can be updated. Right click in the relevant cell in table view, and choose to edit that attribute.

Figure 25.4: Select a sequence list as input to the tool.

In the second wizard step, the source of the attribute information is specified, along with details about how to handle that information.

Figure 25.5: Attributes from 5 columns in the specified file will be added or updated. Existing information will not be overwritten. If one of the specified columns is called TaxID, then a 7-step taxonomy will be downloaded from the NCBI and added to an attribute called Taxonomy.

Attribute information source

- **Attribute file** An Excel file (.xlsx), a comma separated text file (.csv) or a tab separated text file (.tsv) containing attribute information, with a header row containing the names of attributes.

- **Column to match on** The specified column heading will be matched against a sequence attribute name. When a row in that column is identical to the value for that attribute in one or more sequences, the information from the attribute file is added to those sequences. If
there are columns present for attribute types not already defined for the sequence list, that attribute type is added.

- **Include columns** The columns from the attribute file containing attributes to be added to the sequence list. If the "Download taxonomy" option, described below, is checked, a column called Taxonomy will be assumed to included, and will be listed in the preview shown in the next step.

**Configure settings**

- **Overwrite existing information** When checked, if there is a new value for an existing attribute, the old value will be **overwritten** by the new value. When unchecked existing values remain, without change, whether or not a new value is present in the attribute file.

- **Download taxonomy** When checked, a column called TaxIDs is expected, containing valid taxonomic identifiers. A 7-step taxonomy is then downloaded from the NCBI into an attribute called "Taxonomy".

Examples of valid identifiers for TaxID attribute are those found in /db_xref="taxon fields in Genbank entries. For example, for /db_xref="taxon:5833, the expected value in the TaxID column would be 5833.

If a given sequence has an value already set for the Taxonomy attribute, then that existing value remains in place unless the "Overwrite existing information" box is checked.

The next step provides a preview of the updates that will be made. In the upper pane, a list of the attribute types to be considered is listed. For certain attribute types, recognized by particular column names, validation rules are applied. For example, a column named GO-terms is expected to contain terms in the format, GO:<id>, e.g. GO:0046782. For these, the attribute values, as seen in table view, will be hyperlinked to the relevant GO entry online at http://amigo.geneontology.org.

This list of column headings recognized in this way, and how the values in those columns is handled, is described below.

In the bottom pane, attribute values that will be added are shown for a small subset of sequences. If these are not as expected, clicking on the "Previous" button takes you back to the previous step, where the configuration can be updated.

**Column headings and value validation**

Certain column headings are recognized, and if the contents pass validation rules, the entries are handled by the software, generally adding hyperlinks to an online data resources.

Two columns subject to validation have additional handling:

- **TaxID** When valid taxonomic identifiers are found in a TaxID column, and the Download taxonomy checkbox is enabled, then a 7-step taxonomy is then downloaded from the NCBI. This is described further up on this page.

- **Gene ID** The following identifiers in a Gene ID column are added as attribute values and hyperlinked to the relevant online database:
Attributes from several columns are subject to validation checks. If any had failed the check, a yellow exclamation mark in the bottom pane would be shown for that column. Here, all entries pass. The "Other" column is not subject to validation checks. Only one sequence in the list is being updated in this example.

- Ensembl Gene IDs
- HUGO Gene IDs
- VFDB Gene IDs

Any other values in a Gene ID column are added as attributes to the relevant sequences, but are not hyperlinked to an online data resource. Note that this is different to how other non-validated attribute values are handled, as described below.

Multiple identifiers in a given cell, separated by commas, will be added as multiple Gene ID attributes for the relevant sequence. If any one of those identifiers is not recognized as one of the above types, then none will be hyperlinked.

Other columns where contents are validated are those with the headings listed below. If a value in such a column cannot be validated, it is **not** added nor used to update attributes.

If you wish to add information of this type but do not want this level of validation applied, use a heading other than the ones listed below.

- **COG-terms** COG identifiers
- **Compound AROs** Antibiotic Resistance Ontology identifiers
- **Compound Class AROs** Antibiotic Resistance Ontology identifiers
- **Confers-resistance-to ARO** Antibiotic Resistance Ontology identifiers
- **Drug ARO** Antibiotic Resistance Ontology identifiers
- **Drug Class ARO** Antibiotic Resistance Ontology identifiers
- **EC numbers** EC identifiers
- **GenBank accession** Genbank accession numbers
- **Gene ARO** Antibiotic Resistance Ontology identifiers
- **GO-terms** Gene Ontology (GO) identifiers
• **KO-terms** KEGG Orthology (KO) identifiers  
• **Pfam domains** PFAM domain identifiers  
• **Phenotype ARO** Antibiotic Resistance Ontology identifiers  
• **PubMed IDs** Pubmed identifiers  
• **TIGRFAM-terms** TIGRFAM identifiers  
• **Virulence factor ID** Virulence Factors of Pathogenic Bacteria identifiers

**Updating Location-specific attributes**

Location-specific attributes can be created, which are the present for all elements created in that CLC File Location. Such attributes can be updated using the **Update Sequence Attributes in Lists** tool.

Of note when working with such attributes:

- Because these attributes are tied to the Location, they will not appear until the updated sequence list has been saved.
- The updated sequence list must be saved to the same File Location as the input for these attributes and their values to appear.
- If this tool is run on an unsaved sequence list, or using inputs from more than one File Location at the same time, Location-specific attributes will not be updated. Information in the preview pane reflects this.

Location-specific attributes are described in section 3.3.

### 25.4 Split Sequence List

**Split Sequence List** can split up nucleotide or peptide sequence lists. The output can be a specified number of lists, lists containing a specified number of sequences, or lists containing sequences with particular attribute values, such as terms in the description.

To launch the **Split Sequence List** tool, go to:

**Toolbox** | **Utility Tools** ( ) | **Sequence Lists** ( ) | **Split Sequence List** ( )

Select one or more sequence lists as input. If multiple lists are selected, then to process each input list separately, check the "Batch" checkbox. When unchecked, sequences in all the lists are considered together as a single input.

In the next wizard step, the basis upon which to do the splitting and any necessary settings for that option are specified (figure 25.7).

The options are:

- **Split into N lists** In the "Number of lists to create" box, enter the number of lists to split the input into.
• **Create lists with N sequences each** In the "Number of sequences per list" box, enter the relevant number. The final sequence list in the set created may contain fewer than this number.

• **Split based on attribute values** Specify the attribute to split upon from the drop-down list. Columns in the table view of a sequence list equate to the attributes that the list can be split upon.

If no information is entered into the "Attribute values" field, a sequence list is created for each unique value of the specified attribute. If values are provided, a sequence list is created for each of these where at least one sequence has that attribute value. For example, if 3 values are specified, and sequences were found with attributes matching each of these values, 3 sequence lists would be created. If no sequences were found containing 1 of those attribute values, then only 2 sequence lists would be created. Check the "Collect sequences without matches" box to additionally produce a sequence list containing the sequences where no match to a specified value was identified.

Attribute values should be added in a new-line separated list (figure 25.8).

Matching of values is case specific. An asterisk should be added before and/or after the term if the value is part of a longer entry (figure 25.8).

Attribute values are matched in the order they are listed. For example, if the attribute values shown in figure 25.8 were used, and a sequence had a description with both terms in it, that sequence would be placed in the list containing sequences with "Putative" in their descriptions. This is because *Putative* is listed first in the list of values provided to the tool.
Figure 25.8: With the settings shown here, 3 sequence lists were created. These lists are open in the background tabs shown. One contains sequences with descriptions that include the term "Putative", one contains sequences with descriptions that include the term "Uncharacterized", and one contains sequences containing neither term in the description.

25.5 Rename Elements

Using Rename Elements, you can add or remove characters from element names. You can also replace parts of names, optionally using regular expressions to define the sections to replace.

Replacing spaces or other special characters in element names can be particularly useful before exporting to systems where filenames with those characters are problematic to work with.

Other methods of changing or controlling element names exist: Individual elements can be renamed directly in the Navigation Area by doing a slow double-click on the name, or selecting the element and choosing the menu item Edit | Rename. In workflows, the naming of outputs can be configured, as described in section 11.2.3.

To launch the Rename Elements tool, go to:

Toolbox | Utility Tools | Renaming | Rename Elements

In the first wizard step, select the elements to rename (figure 25.9).

Alternatively, you can right-click and select "Add folder contents" or, select "Add folder contents (recursively)" to rename all elements in a folder structure recursively, (figure 25.10).

When multiple elements are supplied as input, each will be renamed. There is no need to check the Batch checkbox for this. However, if you wish to rename only certain elements from the full selection, then checking the Batch box provides the opportunity to do that. When checked, the next wizard step shows the batch overview, where elements can be explicitly included or excluded from those to be renamed, based on text patterns. This step is described in more detail at section 9.3).

Checking the Batch checkbox for this tool also has the following effect when a folder is selected as input:

- With the Batch option checked, the top level contents of that folder will be renamed.
- With the Batch option unchecked, the folder itself will be renamed.
Figure 25.9: Select the elements or folders to be renamed.

Figure 25.10: Right-click for options to add the contents of folders as inputs. Here, the "Add folder contents (recursively)" option was selected. If "Add folder contents" had been selected, only the elements seqlist1 and seqlist2 would have been added to the Selected elements list on the right.

Important notes about renaming elements

- Renaming elements **cannot be undone**. To alter the names further, the elements must be renamed again.

- The renaming action is recorded in the History ( ) for the element, but the "Originates from" entries lists the changed element name, rather than the original element name.

Renaming options

This wizard step presents various options for the renaming action (figure 25.15). The Rename Elements is used for illustration in this section, but the options are the same for the Rename Sequences in Lists tool.
Figure 25.11: Text can be added, removed or replaced in the existing names.

- **Add text to name** Select this option to add text at the beginning or the end of the existing name.

  You can add text directly to these fields, and you can also include placeholders to indicate certain types of information should be added. Multiple placeholders can be used, in combination with other text if desired (figure 25.16). The available placeholders are:

  - `{name}` The current name of the element. Usually used when defining a new naming pattern for replacing the full name of elements.
  - `{shortname}` Truncates the original name to 10 characters. Usually used when replacing the full names of elements.
  - `{Parent folder}` The name of the folder containing the element.
  - `{today}` Today's date in the form YYYY-MM-DD
  - `{enumeration}` Adds a number to the name. This is intended for use when multiple elements are selected as input. Each is assigned a number, starting with the number 1, (added as 0000001) for the first element that was selected, 2 (added as 0000002) for the second element selected, and so on.

  Click in a field and use Shift + F1 (Shift + Fn + F1 on Mac) to show the list of available placeholders, as shown in figure 25.16. Click on a placeholder in that list to have it entered into the field.

- **Shorten name** Select this option to shorten a name by removing a specified number of characters from the start and/or end of the name.

- **Replace part of name** Select this option to specify text or regular expressions to define parts of the element names to be replaced. By default, the text entered in the fields is interpreted literally. Check the "Interpret 'Replace' as regular expression" option to indicate that the terms provided in the "Replace" field should be treated as regular expressions. Information on regular expressions can be found at [http://docs.oracle.com/javase/tutorial/essential/regex/](http://docs.oracle.com/javase/tutorial/essential/regex/).
Figure 25.12: Click on Shift+F1 (Shift + Fn + F1 on Mac) to reveal a drop-down list of placeholders that can be used. Here, today’s date and a hyphen would be prepended, and a hyphen and ascending numeric value appended, to the existing names.

By clicking in either the "Replace" or "with" field and pressing Shift + F1 (Shift + Fn + F1 on Mac), a drop down list of renaming possibilities is presented. The options listed for the Replace field are some commonly used regular expressions. Other standard regular expressions are also admissible in this field. The placeholders described above for adding text to names are available for use in the "with" field. **Note:** We recommend caution when using these placeholders in combination with regular expressions in the Replace field. Please run a small test to ensure it works as you intend.

- **Replace full name** Select this option to replace the full element name. Text and placeholders can be used in this field. The placeholders described above for adding text to names are available for use. Use Shift + F1 (Shift + Fn + F1 on Mac) to see a list.

**Examples using regular expressions to replace parts of element names**

The following are examples to illustrate using regular expressions for renaming. The same principles apply for renaming using the Rename Elements and Rename Sequences in Lists tools.

- Replacing part of an element’s name with today’s date and an underscore. Details are shown in figure 25.17.

- Rename using the first 4 non-whitespace characters from names that start with 2 characters, then have a space, then have multiple characters following, such as 1N R1\_0001.
  - Check the box beside "Interpret ‘Replace’ and ‘with’ as Java regular expressions".
  - Enter ([\w]{2})\s([\w]{2}).* into the "Replaces" field.
  - Enter $1$2 into the "with" field.

- Keep only the last 4 characters of the name.
Figure 25.13: Elements with names Seqlist1 and Seqlist2 each start with a capital letter, followed by 6 small letters. Using the settings shown, their names are updated to be the date the renaming was done, followed by a hyphen, and the remaining parts of the original name, here, the integer at the end of each name.

- Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
- Enter \( (.*)(.{4}$) \) into the "Replaces" field.
- Enter \$2\) into the "with" field.

- Replace a set pattern of text with the name of the parent folder. Here, we start with the name p140101034_1R_AMR and replace the first letter and 9 numbers with the parent folder name.
  
  - Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
  - Enter \(^{\([a-z]\)d(9)}(.*\)\) into the "Replaces" field.
  - Enter \{parentfolder\}\$2\) into the "with" field.

- Rename using just the text between the first and second underscores in 1234_sample-code_5678.
  
  - Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
  - Enter \(^{\[^\_]+(\[^\_]+)((\[^\_]+)}(.*\)\) into the "Replaces" field.
  - Enter \$2\) into the "with" field.

### 25.6 Rename Sequences in Lists

Using Rename Sequences in Lists, you can add or remove characters from the names of sequences within sequence lists. You can also replace parts of sequence names, optionally using regular expressions to define the sections to replace.
Individual sequences in a sequence list can also be renamed manually. Open the sequence list, right-click on the sequence name to change, and select the "Rename Sequence..." option.

To launch the **Rename Sequences in Lists** tool, go to:

**Toolbox | Utility Tools | Renaming | Rename Sequences in Lists**

In the first wizard step, select a sequence list. To rename sequences in multiple sequence lists using the same renaming pattern, check the Batch checkbox (figure 25.14).

![Figure 25.14: When the sequences in more than one list should be renamed, check the Batch checkbox.](image)

The text "Renamed" is added within parentheses to the name of sequence lists output by this tool. E.g. with an input called "seqlist2", the sequence list containing the renamed sequences will be called "seqlist2 (Renamed)".

**Renaming options**

This wizard step presents various options for the renaming action (figure 25.15). The **Rename Elements** is used for illustration in this section, but the options are the same for the **Rename Sequences in Lists** tool.

- **Add text to name** Select this option to add text at the beginning or the end of the existing name.

You can add text directly to these fields, and you can also include placeholders to indicate certain types of information should be added. Multiple placeholders can be used, in combination with other text if desired (figure 25.16). The available placeholders are:

  - `{name}` The current name of the element. Usually used when defining a new naming pattern for replacing the full name of elements.
  - `{shortname}` Truncates the original name to 10 characters. Usually used when replacing the full names of elements.
  - `{Parent folder}` The name of the folder containing the element.
  - `{today}` Today's date in the form YYYY-MM-DD
  - `{enumeration}` Adds a number to the name. This is intended for use when multiple elements are selected as input. Each is assigned a number, starting with the number 1, (added as 0000001) for the first element that was selected, 2 (added as 0000002) for the second element selected, and so on.
Figure 25.15: Text can be added, removed or replaced in the existing names.

Click in a field and use Shift + F1 (Shift + Fn + F1 on Mac) to show the list of available placeholders, as shown in figure 25.16. Click on a placeholder in that list to have it entered into the field.

Figure 25.16: Click on Shift+F1 (Shift + Fn + F1 on Mac) to reveal a drop-down list of placeholders that can be used. Here, today's date and a hyphen would be prepended, and a hyphen and ascending numeric value appended, to the existing names.

- **Shorten name** Select this option to shorten a name by removing a specified number of characters from the start and/or end of the name.

- **Replace part of name** Select this option to specify text or regular expressions to define parts of the element names to be replaced. By default, the text entered in the fields is interpreted literally. Check the "Interpret 'Replace' as regular expression" option to indicate that the terms provided in the "Replace" field should be treated as regular
expressions. Information on regular expressions can be found at http://docs.oracle.com/javase/tutorial/essential/regex/.

By clicking in either the "Replace" or "with" field and pressing Shift + F1 (Shift + Fn + F1 on Mac), a drop down list of renaming possibilities is presented. The options listed for the Replace field are some commonly used regular expressions. Other standard regular expressions are also admissible in this field. The placeholders described above for adding text to names are available for use in the "with" field. **Note:** We recommend caution when using these placeholders in combination with regular expressions in the Replace field. Please run a small test to ensure it works as you intend.

- **Replace full name** Select this option to replace the full element name. Text and placeholders can be used in this field. The placeholders described above for adding text to names are available for use. Use Shift + F1 (Shift + Fn + F1 on Mac) to see a list.

**Examples using regular expressions to replace parts of element names**

The following are examples to illustrate using regular expressions for renaming. The same principles apply for renaming using the **Rename Elements** and **Rename Sequences in Lists** tools.

- Replacing part of an element’s name with today’s date and an underscore. Details are shown in figure 25.17.

![Figure 25.17: Elements with names Seqlist1 and Seqlist2 each start with a capital letter, followed by 6 small letters. Using the settings shown, their names are updated to be the date the renaming was done, followed by a hyphen, and the remaining parts of the original name, here, the integer at the end of each name.](image)

- Rename using the first 4 non-whitespace characters from names that start with 2 characters, then have a space, then have multiple characters following, such as 1N R1\_0001.
– Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
– Enter ([\w]{2})\s([\w]{2}).* into the "Replaces" field.
– Enter $1$2 into the "with" field.

• Keep only the last 4 characters of the name.
  – Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
  – Enter (.\*) (.\{4}\$) into the "Replaces" field.
  – Enter $2$ into the "with" field.

• Replace a set pattern of text with the name of the parent folder. Here, we start with the name p140101034_1R_AMR and replace the first letter and 9 numbers with the parent folder name.
  – Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
  – Enter ([a-z]\d{9})(.*) into the "Replaces" field.
  – Enter {parentfolder}$2$ into the "with" field.

• Rename using just the text between the first and second underscores in 1234_sample-code_5678.
  – Check the box beside "Interpret 'Replace' and 'with' as Java regular expressions".
  – Enter (^[^\_]+)_([^\_]+)(.*) into the "Replaces" field.
  – Enter $2$ into the "with" field.
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.
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
- **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.

Certain types of graphs can additionally contain the following settings:

- **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 representation of the data is configured in the bottom area, e.g. line widths, dot types, colors, etc. For graphs of multiple data series, the series to apply the settings to can be selected from a drop down list.
• **Dot type** Can be None, Cross, Plus, Square, Diamond, Circle, Triangle, Reverse triangle, or Dot.

• **Dot color** Click the color box to select a color.

• **Line width** Thin, Medium or Wide

• **Line type** None, Line, Long dash or Short dash

• **Line color** Click the color box to select a color.

The graph and 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.
- **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/](http://www.rcsb.org/pdb/).
- **env_nr.** Non-redundant CDS translations from env_nt entries.
- **tsa_nr.** Transcriptome Shotgun Assembly db proteins. (NCBI BLAST only)
- **month.** All new or revised GenBank CDS translations + PDB + SwissProt + PIR + PRF released in the last 30 days. (Create Protein Report only)

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.
- **Human G+T.** Human genomic and transcript sequences
- **Mouse G+T.** Mouse genomic and transcript sequences
- **refseq_rna.** mRNA sequences from NCBI Reference Sequence Project.
• **refseq_genomic.** Genomic sequences from NCBI Reference Sequence Project.

• **refseq_representative_genomes.** Representative 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.

• **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.

• **env\_nt.** Sequences from environmental samples, such as uncultured bacterial samples isolated from soil or marine samples. The largest single source is Sagarsso Sea project. This does overlap with nucleotide nr.

• **tsa\_nt.** Transcriptome Shotgun Assembly database.

• **prokaryotic\_16S\_ribosomal\_RNA.** 16S ribosomal RNA sequences.

• **Betacoronavirus.** NCBI database of betacoronavirus sequences.

### 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.


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**: [https://resources.qiagenbioinformatics.com/wbsettings/NCBI_BlastNucleotideDatabases.zip](https://resources.qiagenbioinformatics.com/wbsettings/NCBI_BlastNucleotideDatabases.zip)

- **Protein databases**: [https://resources.qiagenbioinformatics.com/wbsettings/NCBI_BlastProteinDatabases.zip](https://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:

```plaintext
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](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.
<table>
<thead>
<tr>
<th>Name</th>
<th>P4</th>
<th>P3</th>
<th>P2</th>
<th>P1</th>
<th>P1’</th>
<th>P2’</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyanogen bromide (CNBr)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Asp-N endopeptidase</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>D</td>
<td>-</td>
</tr>
<tr>
<td>Arg-C</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>R</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Lys-C</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>K</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>K, R</td>
<td>not P</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin</td>
<td>-</td>
<td>-</td>
<td>W</td>
<td>K</td>
<td>P</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin</td>
<td>-</td>
<td>-</td>
<td>M</td>
<td>R</td>
<td>P</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin*</td>
<td>-</td>
<td>-</td>
<td>C, D</td>
<td>K</td>
<td>D</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin*</td>
<td>-</td>
<td>-</td>
<td>C</td>
<td>K</td>
<td>H, Y</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin*</td>
<td>-</td>
<td>-</td>
<td>C</td>
<td>R</td>
<td>K</td>
<td>-</td>
</tr>
<tr>
<td>Trypsin*</td>
<td>-</td>
<td>-</td>
<td>R</td>
<td>R</td>
<td>H,R</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-high spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>F, Y</td>
<td>not P</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-high spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>W</td>
<td>not M, P</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-low spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>F, L, Y</td>
<td>not P</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-low spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>W</td>
<td>not M, P</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-low spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>M</td>
<td>not P, Y</td>
<td>-</td>
</tr>
<tr>
<td>Chymotrypsin-low spec.</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>H</td>
<td>not D, M, P, W</td>
<td>-</td>
</tr>
<tr>
<td>o-Iodosobenzoate</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>W</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Thermolysin</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>not D, E</td>
<td>A, F, I, L, M or V</td>
<td>-</td>
</tr>
<tr>
<td>Post-Pro</td>
<td>-</td>
<td>-</td>
<td>H, K, R</td>
<td>P</td>
<td>not P</td>
<td>-</td>
</tr>
<tr>
<td>Glu-C</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>E</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Asp-N</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>D</td>
<td>-</td>
</tr>
<tr>
<td>Proteinase K</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>A, E, F, I, L, T, V, W, Y</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Factor Xa</td>
<td>A, F, G, I, L, T, V, M</td>
<td>D,E</td>
<td>G</td>
<td>R</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Granzyme B</td>
<td>I</td>
<td>E</td>
<td>P</td>
<td>D</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Thrombin</td>
<td>-</td>
<td>-</td>
<td>G</td>
<td>R</td>
<td>G</td>
<td>-</td>
</tr>
<tr>
<td>TEV (Tobacco Etch Virus)</td>
<td>-</td>
<td>Y</td>
<td>-</td>
<td>Q</td>
<td>G, S</td>
<td>-</td>
</tr>
</tbody>
</table>
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: https://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 to 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 https://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).

![Gateway cloning configuration selection](image-url)

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

<table>
<thead>
<tr>
<th>One-letter abbreviation</th>
<th>Three-letter abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Ala</td>
<td>Alanine</td>
</tr>
<tr>
<td>R</td>
<td>Arg</td>
<td>Arginine</td>
</tr>
<tr>
<td>N</td>
<td>Asn</td>
<td>Asparagine</td>
</tr>
<tr>
<td>D</td>
<td>Asp</td>
<td>Aspartic acid</td>
</tr>
<tr>
<td>C</td>
<td>Cys</td>
<td>Cysteine</td>
</tr>
<tr>
<td>Q</td>
<td>Gln</td>
<td>Glutamine</td>
</tr>
<tr>
<td>E</td>
<td>Glu</td>
<td>Glutamic acid</td>
</tr>
<tr>
<td>G</td>
<td>Gly</td>
<td>Glycine</td>
</tr>
<tr>
<td>H</td>
<td>His</td>
<td>Histidine</td>
</tr>
<tr>
<td>J</td>
<td>Xle</td>
<td>Leucine or Isoleucine</td>
</tr>
<tr>
<td>L</td>
<td>Leu</td>
<td>Leucine</td>
</tr>
<tr>
<td>I</td>
<td>ILe</td>
<td>Isoleucine</td>
</tr>
<tr>
<td>K</td>
<td>Lys</td>
<td>Lysine</td>
</tr>
<tr>
<td>M</td>
<td>Met</td>
<td>Methionine</td>
</tr>
<tr>
<td>F</td>
<td>Phe</td>
<td>Phenylalanine</td>
</tr>
<tr>
<td>P</td>
<td>Pro</td>
<td>Proline</td>
</tr>
<tr>
<td>O</td>
<td>Pyl</td>
<td>Pyrrolysine</td>
</tr>
<tr>
<td>U</td>
<td>Sec</td>
<td>Selenocysteine</td>
</tr>
<tr>
<td>S</td>
<td>Ser</td>
<td>Serine</td>
</tr>
<tr>
<td>T</td>
<td>Thr</td>
<td>Threonine</td>
</tr>
<tr>
<td>W</td>
<td>Trp</td>
<td>Tryptophan</td>
</tr>
<tr>
<td>Y</td>
<td>Tyr</td>
<td>Tyrosine</td>
</tr>
<tr>
<td>V</td>
<td>Val</td>
<td>Valine</td>
</tr>
<tr>
<td>B</td>
<td>Asx</td>
<td>Aspartic acid or Asparagine</td>
</tr>
<tr>
<td>Z</td>
<td>Glx</td>
<td>Glutamic acid or Glutamine</td>
</tr>
<tr>
<td>X</td>
<td>Xaa</td>
<td>Any amino acid</td>
</tr>
</tbody>
</table>
Appendix G

IUPAC codes for nucleotides

(Single-letter codes based on International Union of Pure and Applied Chemistry)


<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Adenine</td>
</tr>
<tr>
<td>C</td>
<td>Cytosine</td>
</tr>
<tr>
<td>G</td>
<td>Guanine</td>
</tr>
<tr>
<td>T</td>
<td>Thymine</td>
</tr>
<tr>
<td>U</td>
<td>Uracil</td>
</tr>
<tr>
<td>R</td>
<td>Purine (A or G)</td>
</tr>
<tr>
<td>Y</td>
<td>Pyrimidine (C, T, or U)</td>
</tr>
<tr>
<td>M</td>
<td>C or A</td>
</tr>
<tr>
<td>K</td>
<td>T, U, or G</td>
</tr>
<tr>
<td>W</td>
<td>T, U, or A</td>
</tr>
<tr>
<td>S</td>
<td>C or G</td>
</tr>
<tr>
<td>B</td>
<td>C, T, U, or G (not A)</td>
</tr>
<tr>
<td>D</td>
<td>A, T, U, or G (not C)</td>
</tr>
<tr>
<td>H</td>
<td>A, T, U, or C (not G)</td>
</tr>
<tr>
<td>V</td>
<td>A, C, or G (not T, not U)</td>
</tr>
<tr>
<td>N</td>
<td>Any base (A, C, G, T, or U)</td>
</tr>
</tbody>
</table>
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, trees, etc.
### H.1.1 Sequence data formats

<table>
<thead>
<tr>
<th><strong>File type</strong></th>
<th><strong>Suffix</strong></th>
<th><strong>Import</strong></th>
<th><strong>Export</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>AB1</td>
<td>.ab1</td>
<td>X</td>
<td></td>
<td>Including chromatograms</td>
</tr>
<tr>
<td>ABI</td>
<td>.abi</td>
<td>X</td>
<td></td>
<td>Including chromatograms</td>
</tr>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>Clone manager</td>
<td>.cm5</td>
<td>X</td>
<td></td>
<td>Clone manager sequence format</td>
</tr>
<tr>
<td>DNAstrider</td>
<td>.str/.strider</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>DS Gene</td>
<td>.bsml</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EMBL</td>
<td>.emb/.embl</td>
<td>X</td>
<td>X</td>
<td>Rich information incl. annotations (nucs only)</td>
</tr>
<tr>
<td>FASTA</td>
<td>.fa/.fsa/.fasta</td>
<td>X</td>
<td>X</td>
<td>Simple format, name &amp; description</td>
</tr>
<tr>
<td>GenBank</td>
<td>.gbk/.gb/.gp/.gbff</td>
<td>X</td>
<td></td>
<td>Rich information incl. annotations</td>
</tr>
<tr>
<td>Gene Construction Kit</td>
<td>.gck</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lasergene</td>
<td>.pro/.seq</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nexus</td>
<td>.nxs/.nexus</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Phred</td>
<td>.phd</td>
<td>X</td>
<td></td>
<td>Including chromatograms</td>
</tr>
<tr>
<td>PIR (NBRF)</td>
<td>.pir</td>
<td>X</td>
<td>X</td>
<td>Simple format, name &amp; description</td>
</tr>
<tr>
<td>Raw sequence</td>
<td>any</td>
<td>X</td>
<td></td>
<td>Only sequence (no name)</td>
</tr>
<tr>
<td>SCF2</td>
<td>.scf</td>
<td>X</td>
<td></td>
<td>Including chromatograms</td>
</tr>
<tr>
<td>SCF3</td>
<td>.scf</td>
<td>X</td>
<td>X</td>
<td>Including chromatograms</td>
</tr>
<tr>
<td>Sequence Comma separated values</td>
<td>.csv</td>
<td>X</td>
<td>X</td>
<td>Simple format. One seq per line: name, description(optional), sequence</td>
</tr>
<tr>
<td>Staden</td>
<td>.sdn</td>
<td>X</td>
<td></td>
<td>Rich information incl. annotations (only peptides)</td>
</tr>
<tr>
<td>Swiss-Prot</td>
<td>.swp</td>
<td>X</td>
<td>X</td>
<td>Rich information incl. annotations</td>
</tr>
<tr>
<td>Tab delimited text</td>
<td>.txt</td>
<td>X</td>
<td></td>
<td>Annotations in tab delimited text format</td>
</tr>
<tr>
<td>Vector NTI archives*</td>
<td>.ma4/.pa4/.oa4</td>
<td>X</td>
<td></td>
<td>Archives in rich format</td>
</tr>
<tr>
<td>Vector NTI Database*</td>
<td></td>
<td>X</td>
<td></td>
<td>Special import full database</td>
</tr>
</tbody>
</table>

When exporting in fasta format, it is possible to remove sequence ends covered by annotations of type "Trim" (read more in section 19.2).
### H.1.2 Contig formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE</td>
<td>.ace</td>
<td>X</td>
<td>X</td>
<td>No chromatogram or quality score</td>
</tr>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
</tbody>
</table>

### H.1.3 Alignment formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aligned fasta</td>
<td>.fa/.fsa/.fasta</td>
<td>X</td>
<td>X</td>
<td>Simple fasta-based format with – for gaps</td>
</tr>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>ClustalW</td>
<td>.aln</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>GCG Alignment</td>
<td>.msf</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Nexus</td>
<td>.nxs/.nexus</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Phylip Alignment</td>
<td>.phy</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

### H.1.4 Tree formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>Newick</td>
<td>.nwk</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Nexus</td>
<td>.nxs/.nexus</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

### H.1.5 Expression data formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affymetrix CHP</td>
<td>.chp/.psi</td>
<td>X</td>
<td></td>
<td>Expression values and annotations, see section I.2.</td>
</tr>
<tr>
<td>Affymetrix pivot/metric</td>
<td>.txt/.csv</td>
<td>X</td>
<td></td>
<td>Gene-level expression value, see section I.2.</td>
</tr>
<tr>
<td>Affymetrix NetAffx</td>
<td>.csv</td>
<td>X</td>
<td></td>
<td>Annotations, see section I.2.</td>
</tr>
<tr>
<td>BED</td>
<td>.bed</td>
<td></td>
<td>X</td>
<td>Expression values, see section ??</td>
</tr>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>Excel</td>
<td>.xls/.xlsx</td>
<td></td>
<td>X</td>
<td>All tables and reports</td>
</tr>
<tr>
<td>Generic</td>
<td>.txt/.csv</td>
<td>X</td>
<td></td>
<td>Expression values</td>
</tr>
<tr>
<td>Generic</td>
<td>.txt/.csv</td>
<td></td>
<td>X</td>
<td>Annotations</td>
</tr>
<tr>
<td>GEO sample/series</td>
<td>.txt/.csv</td>
<td>X</td>
<td></td>
<td>Expression values, see section I.1</td>
</tr>
<tr>
<td>Illumina</td>
<td>.txt</td>
<td></td>
<td>X</td>
<td>Expression values and annotations, see section I.3</td>
</tr>
<tr>
<td>Table CSV</td>
<td>.csv</td>
<td></td>
<td>X</td>
<td>Samples and experiments, see section I.5</td>
</tr>
<tr>
<td>Tab delimited</td>
<td>.txt</td>
<td>X</td>
<td>X</td>
<td>Samples and experiments, see section I.5</td>
</tr>
</tbody>
</table>
### H.1.6 Other formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>PDB</td>
<td>.pdb</td>
<td>X</td>
<td></td>
<td>3D structure</td>
</tr>
<tr>
<td>RNA structures</td>
<td>.ct, .col, .maml/.xml</td>
<td></td>
<td>X</td>
<td>Secondary structure for RNA</td>
</tr>
</tbody>
</table>

### H.1.7 Table and text formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excel</td>
<td>.xls/.xlsx</td>
<td>X</td>
<td>X</td>
<td>All tables and reports</td>
</tr>
<tr>
<td>Table CSV</td>
<td>.csv</td>
<td>X</td>
<td>X</td>
<td>All tables</td>
</tr>
<tr>
<td>Tab delimited</td>
<td>.tsv</td>
<td></td>
<td>X</td>
<td>All tables</td>
</tr>
<tr>
<td>Tab delimited</td>
<td>.txt</td>
<td></td>
<td></td>
<td>All tables</td>
</tr>
<tr>
<td>Text</td>
<td>.txt</td>
<td>X</td>
<td>X</td>
<td>All data in a textual format</td>
</tr>
<tr>
<td>CLC</td>
<td>.clc</td>
<td>X</td>
<td>X</td>
<td>Rich format including all information</td>
</tr>
<tr>
<td>HTML</td>
<td>.html</td>
<td></td>
<td>X</td>
<td>All tables</td>
</tr>
<tr>
<td>PDF</td>
<td>.pdf</td>
<td></td>
<td></td>
<td>Export reports in Portable Document Format</td>
</tr>
</tbody>
</table>

Please see section H.1.5 for special cases of table imports.

### H.1.8 File compression formats

<table>
<thead>
<tr>
<th>File type</th>
<th>Suffix</th>
<th>Import</th>
<th>Export</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zip export</td>
<td>.zip</td>
<td></td>
<td>X</td>
<td>Selected files in CLC format</td>
</tr>
<tr>
<td>Zip import</td>
<td>.zip/.gz/.tar</td>
<td>X</td>
<td></td>
<td>Contained files/folder structure (.tar and .zip not supported for NGS data)</td>
</tr>
</tbody>
</table>

**Note:** It is possible to import 'external' files into the CLC Main Workbench and view these in the Navigation Area, but it is only the above mentioned formats whose contents can be shown in the CLC Main 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 7.3 for further details).

<table>
<thead>
<tr>
<th>Format</th>
<th>Suffix</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portable Network Graphics</td>
<td>.png</td>
<td>bitmap</td>
</tr>
<tr>
<td>JPEG</td>
<td>.jpg</td>
<td>bitmap</td>
</tr>
<tr>
<td>Tagged Image File</td>
<td>.tif</td>
<td>bitmap</td>
</tr>
<tr>
<td>PostScript</td>
<td>.ps</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Encapsulated PostScript</td>
<td>.eps</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Portable Document Format</td>
<td>.pdf</td>
<td>vector graphics</td>
</tr>
<tr>
<td>Scalable Vector Graphics</td>
<td>.svg</td>
<td>vector graphics</td>
</tr>
</tbody>
</table>
Appendix I

Gene expression annotation files and microarray data formats

The CLC Main 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 §Generic expression and annotation data file formats).

The CLC Main Workbench assumes that expression values are given at the gene level, thus probe-level analysis of Affymetrix GeneChips and import of Affymetrix CEL and CDF files is currently not supported. However, the CLC Main Workbench allows import of txt files exported from R containing processed Affymetrix CEL-file data (see §Affymetrix GeneChip).

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 §Generic expression and annotation data file formats).

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
```
Figure I.1: Selecting Samples, SOFT and Data before clicking go will give you the format supported by the CLC Main Workbench.

The first line should start with ^SAMPLE = followed by the sample name, the line !sample_table_begin and the line !sample_table_end. Between the !sample_table_begin and !sample_table_end, lines are the column contents of the sample.

Note that GEO sample importer will also work for concatenated GEO sample files — allowing multiple samples to be imported in one go. Download a sample file containing concatenated sample files here: https://resources.qiagenbioinformatics.com/madata/GEOSampleFilesConcatenated.txt

Below you can find examples of the formatting of the GEO formats.

**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: https://resources.qiagenbioinformatics.com/madata/GEOSampleFileSimple.txt

**GEO sample file, including present/absent calls**

This format includes an extra column for absent/present calls that can also be imported.

```
^SAMPLE = GSM21610
```
APPENDIX I. GENE EXPRESSION ANNOTATION FILES AND MICROARRAY DATA FORMATS

!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:
https://resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresent.txt

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:
https://resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresentCallAndPValue.txt

GEO sample file: using absent/present call and p-value columns for sequence information

The CLC Main 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
!sample_table_end
### APPENDIX I. GENE EXPRESSION ANNOTATION FILES AND MICROARRAY DATA FORMATS

```
id3      50.4      ATA
id4      57.8      ATT
id5    2914.1      TTA
!sample_table_end
```

Download the sample file here: [https://resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequenceTag.txt](https://resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequenceTag.txt)

Or, if you have multiple probes per sequence you could use the present/absent column to hold the sequence name and the p-value column to hold the interrogation position of your probes:

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF    VALUE    ABS_CALL    DETECTION    P-VALUE
probe1    755.07    seq1       1452         
probe2    587.88    seq1       497          
probe3    716.29    seq1       1447         
probe4    1287.18   seq2       1899         
!sample_table_end
```

Download the sample file here: [https://resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequenceTagAndProbe.txt](https://resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequenceTagAndProbe.txt)

### 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: [https://resources.qiagenbioinformatics.com/madata/GEOSeriesFile.txt](https://resources.qiagenbioinformatics.com/madata/GEOSeriesFile.txt)

### 1.2 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 log-transformed expression values in three lines of code:

```r
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 log2 expression levels to 'evals.txt'
```

The exported txt file (evals.txt) can be imported into the CLC Main Workbench using the Generic expression data table format importer (see internalrefsec:customexpressiondataformatssectionGeneric expression and annotation data file formats; 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.

**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):
https://resources.qiagenbioinformatics.com/madata/AffymetrixCHPandPSI.zip

**Affymetrix metrix files**

The Affymetrix metrics or pivot files are tab-delimited files that may be exported from the Affymetrix scanner software. The metrics files have a lot of technical information that is only partly used in the CLC Main 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 CLC Main Workbench.

Download a small example sample file here:
https://resources.qiagenbioinformatics.com/madata/AffymetrixMetrics.txt
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 23.1.3.

Download a small example annotation file here which includes header information: [https://resources.qiagenbioinformatics.com/madata/AffymetrixNetAffxAnnotationFile.csv](https://resources.qiagenbioinformatics.com/madata/AffymetrixNetAffxAnnotationFile.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.

Illumina expression data, compact format

An example of this format is shown below:

<table>
<thead>
<tr>
<th>TargetID</th>
<th>AVG_Signal</th>
<th>BEAD_STDEV</th>
<th>Detection</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI_10047089-S</td>
<td>112.5</td>
<td>4.2</td>
<td>0.16903226</td>
</tr>
<tr>
<td>GI_10047091-S</td>
<td>127.6</td>
<td>4.8</td>
<td>0.76774194</td>
</tr>
</tbody>
</table>

All this information is imported into the CLC Main Workbench. The AVG_Signal is used as the expression measure.

Download a small sample file here: [https://resources.qiagenbioinformatics.com/madata/IlluminaBeadChipCompact.txt](https://resources.qiagenbioinformatics.com/madata/IlluminaBeadChipCompact.txt)

Illumina expression data, extended format

An example of this format is shown below:

<table>
<thead>
<tr>
<th>TargetID</th>
<th>MIN_Signal</th>
<th>AVG_Signal</th>
<th>MAX_Signal</th>
<th>NARRAYS</th>
<th>ARRAY_STDEV</th>
<th>BEAD_STDEV</th>
<th>Avg_NBEADS</th>
<th>Detection</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI_10047089-S</td>
<td>73.7</td>
<td>73.7</td>
<td>73.7</td>
<td>1</td>
<td>NaN</td>
<td>3.4</td>
<td>53</td>
<td>0.05669084</td>
</tr>
<tr>
<td>GI_10047091-S</td>
<td>312.7</td>
<td>312.7</td>
<td>312.7</td>
<td>1</td>
<td>NaN</td>
<td>11.1</td>
<td>50</td>
<td>0.99604483</td>
</tr>
</tbody>
</table>

All this information is imported into the CLC Main Workbench. The AVG_Signal is used as the expression measure.

Download a small sample file here: [https://resources.qiagenbioinformatics.com/madata/IlluminaBeadChipExtended.txt](https://resources.qiagenbioinformatics.com/madata/IlluminaBeadChipExtended.txt)

Illumina expression data, with annotations

An example of this format is shown below:

<table>
<thead>
<tr>
<th>TargetID</th>
<th>Accession</th>
<th>Symbol</th>
<th>Definition</th>
<th>Synonym</th>
<th>Signal-BG02</th>
<th>DCp32</th>
<th>Detection-BG02</th>
<th>DCp32</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI_10047089-S</td>
<td>NM_014332.1</td>
<td>SMPX</td>
<td>&quot;Homo sapiens small muscle protein, X-linked (SMPX), mRNA.&quot;</td>
<td>-17.6</td>
<td>0.03559657</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GI_10047091-S</td>
<td>NM_013259.1</td>
<td>NP25</td>
<td>&quot;Homo sapiens neuronal protein (NP25), mRNA.&quot;</td>
<td>NP22</td>
<td>32.6</td>
<td>0.99604483</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GI_10047093-S</td>
<td>NM_016299.1</td>
<td>HSP70-4</td>
<td>&quot;Homo sapiens likely ortholog of mouse heat shock protein, 70 kDa 4 (HSP70-4), mRNA.&quot;</td>
<td>228.1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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:
https://resources.qiagenbioinformatics.com/madata/IlluminaBeadStudioWithAnnotations.txt

### 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:
https://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 **CLC Main Workbench** recognizes the 18 samples and their columns.

### Illumina annotation files

The **CLC Main 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 23.1.3.

Download a small example annotation file of the first type here:
https://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 23.1.3. 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.


### 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.

**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 this example as a file here: [https://resources.qiagenbioinformatics.com/madata/CustomExpressionData.txt](https://resources.qiagenbioinformatics.com/madata/CustomExpressionData.txt)

**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’, ‘ProbelID’ 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:

[https://resources.qiagenbioinformatics.com/madata/SimpleCustomAnnotation.csv](https://resources.qiagenbioinformatics.com/madata/SimpleCustomAnnotation.csv)

[https://resources.qiagenbioinformatics.com/madata/FullCustomAnnotation.csv](https://resources.qiagenbioinformatics.com/madata/FullCustomAnnotation.csv)

To meet requirements imposed by special functionalities in the **CLC Main 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:

<table>
<thead>
<tr>
<th>Column header in imported file (alternatives separated by commas)</th>
<th>Label in experiment table</th>
<th>Description (tool tip)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe Set ID, Feature ID, ProbeID, Probe_Id, transcript_cluster_id</td>
<td>Feature ID</td>
<td>Probe identifier tag</td>
</tr>
<tr>
<td>Representative Public ID, Public identifier tag, GenbankAccession</td>
<td>Public identifier tag</td>
<td>Representative public ID</td>
</tr>
<tr>
<td>Gene Symbol, GeneSymbol</td>
<td>Gene symbol</td>
<td>Gene symbol</td>
</tr>
<tr>
<td>Gene Ontology Biological Process, Ontology_Process, GO_biological_process</td>
<td>GO biological process</td>
<td>Gene Ontology biological process</td>
</tr>
<tr>
<td>Gene Ontology Cellular Component, Ontology_Component, GO_cellular_component</td>
<td>GO cellular component</td>
<td>Gene Ontology cellular component</td>
</tr>
<tr>
<td>Gene Ontology Molecular Function, Ontology_Function, GO_molecular_function</td>
<td>GO molecular function</td>
<td>Gene Ontology molecular function</td>
</tr>
<tr>
<td>Pathway</td>
<td>Pathway</td>
<td>Pathway</td>
</tr>
</tbody>
</table>

The full list of possible column headers:
<table>
<thead>
<tr>
<th>Column header in imported file (alternatives separated by commas)</th>
<th>Label in experiment table</th>
<th>Description (tool tip)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Species Scientific Name, Species Name, Species</td>
<td>Species name</td>
<td>Scientific species name</td>
</tr>
<tr>
<td>GeneChip Array</td>
<td>Gene chip array</td>
<td>Gene Chip Array name</td>
</tr>
<tr>
<td>Annotation Date</td>
<td>Annotation date</td>
<td>Date of annotation</td>
</tr>
<tr>
<td>Sequence Type</td>
<td>Sequence type</td>
<td>Type of sequence</td>
</tr>
<tr>
<td>Sequence Source</td>
<td>Sequence source</td>
<td>Source from which sequence was obtained</td>
</tr>
<tr>
<td>Transcript ID (Array Design), Transcript</td>
<td>transcript Id</td>
<td>Transcript identifier tag</td>
</tr>
<tr>
<td>Target Description</td>
<td>Target description</td>
<td>Target description</td>
</tr>
<tr>
<td>Archival UniGene Cluster</td>
<td>Archival UniGene cluster</td>
<td>Archival UniGene cluster</td>
</tr>
<tr>
<td>UniGene ID, UniGeneId, Unigene_ID, unigene</td>
<td>UniGene ID</td>
<td>UniGene identifier tag</td>
</tr>
<tr>
<td>Genome Version</td>
<td>Genome version</td>
<td>Version of genome on which annotation is based</td>
</tr>
<tr>
<td>Alignments</td>
<td>Alignments</td>
<td>Alignments</td>
</tr>
<tr>
<td>Gene Title</td>
<td>Gene title</td>
<td>Gene title</td>
</tr>
<tr>
<td>geng_assignments</td>
<td>geng_assignments</td>
<td>geng_assignments</td>
</tr>
<tr>
<td>Chromosomal Location</td>
<td>Chromosomal location</td>
<td>Chromosomal location</td>
</tr>
<tr>
<td>Unigene Cluster Type</td>
<td>Unigene Cluster type</td>
<td>Unigene cluster type</td>
</tr>
<tr>
<td>Ensemble</td>
<td>Ensemble</td>
<td>Ensemble</td>
</tr>
<tr>
<td>Entrez Gene, EntrezGeneID, Entrez_Gene_ID</td>
<td>Entrez gene</td>
<td>Entrez gene</td>
</tr>
<tr>
<td>SwissProt</td>
<td>SwissProt</td>
<td>SwissProt</td>
</tr>
<tr>
<td>EC</td>
<td>EC</td>
<td>EC</td>
</tr>
<tr>
<td>OMIM</td>
<td>OMIM</td>
<td>Online Mendelian Inheritance in Man</td>
</tr>
<tr>
<td>RefSeq Protein ID</td>
<td>RefSeq protein ID</td>
<td>RefSeq protein identifier tag</td>
</tr>
<tr>
<td>RefSeq Transcript ID</td>
<td>RefSeq transcript ID</td>
<td>RefSeq transcript identifier tag</td>
</tr>
<tr>
<td>FlyBase</td>
<td>FlyBase</td>
<td>FlyBase</td>
</tr>
<tr>
<td>AGI</td>
<td>AGI</td>
<td>AGI</td>
</tr>
<tr>
<td>WormBase</td>
<td>WormBase</td>
<td>WormBase</td>
</tr>
<tr>
<td>MGI Name</td>
<td>MGI name</td>
<td>MGI name</td>
</tr>
<tr>
<td>RGD Name</td>
<td>RGD name</td>
<td>RGD name</td>
</tr>
<tr>
<td>SGD accession number</td>
<td>SGD accession number</td>
<td>SGD accession number</td>
</tr>
<tr>
<td>InterPro</td>
<td>InterPro</td>
<td>InterPro</td>
</tr>
<tr>
<td>Transcript Assignments</td>
<td>Transcript assignments</td>
<td>Transcript assignments</td>
</tr>
<tr>
<td>mRNA assignments</td>
<td>mRNA assignments</td>
<td>mRNA assignments</td>
</tr>
<tr>
<td>Annotation Notes</td>
<td>Annotation notes</td>
<td>Annotation notes</td>
</tr>
<tr>
<td>GO, Ontology</td>
<td>GO, Ontology</td>
<td>GO, Ontology</td>
</tr>
<tr>
<td>Cytoband</td>
<td>Cytoband</td>
<td>Cytoband</td>
</tr>
<tr>
<td>PrimaryAccession</td>
<td>Primary accession</td>
<td>Primary accession</td>
</tr>
<tr>
<td>RefSeqAccession</td>
<td>RefSeq accession</td>
<td>RefSeq accession</td>
</tr>
<tr>
<td>Gene Name</td>
<td>Gene name</td>
<td>Gene name</td>
</tr>
<tr>
<td>TIGRID</td>
<td>TIGRID</td>
<td>TIGRID</td>
</tr>
<tr>
<td>Description</td>
<td>Description</td>
<td>Description</td>
</tr>
<tr>
<td>GenomicCoordinates</td>
<td>Genomic coordinates</td>
<td>Genomic coordinates</td>
</tr>
<tr>
<td>Search key</td>
<td>Search key</td>
<td>Search key</td>
</tr>
<tr>
<td>Target</td>
<td>Target</td>
<td>Target</td>
</tr>
<tr>
<td>Genbank identifier</td>
<td>Genbank identifier</td>
<td>Genbank identifier</td>
</tr>
<tr>
<td>GenBank accession</td>
<td>GenBank accession</td>
<td>GenBank accession</td>
</tr>
<tr>
<td>Gene symbol</td>
<td>Gene symbol</td>
<td>Gene symbol</td>
</tr>
<tr>
<td>Probe type</td>
<td>Probe type</td>
<td>Probe type</td>
</tr>
<tr>
<td>Cossahyb_type</td>
<td>Cossahyb type</td>
<td>Cossahyb type</td>
</tr>
<tr>
<td>category</td>
<td>category</td>
<td>category</td>
</tr>
<tr>
<td>Start, Probe_Start</td>
<td>Start</td>
<td>Start</td>
</tr>
<tr>
<td>Stop</td>
<td>Stop</td>
<td>Stop</td>
</tr>
<tr>
<td>Definition</td>
<td>Definition</td>
<td>Definition</td>
</tr>
<tr>
<td>Synonym, Synonyms</td>
<td>Synonym</td>
<td>Synonym</td>
</tr>
<tr>
<td>Source</td>
<td>Source</td>
<td>Source</td>
</tr>
<tr>
<td>Source Reference ID</td>
<td>Source Reference ID</td>
<td>Source Reference ID</td>
</tr>
<tr>
<td>RefSeqs_ID</td>
<td>Reference sequence id</td>
<td>Reference sequence id</td>
</tr>
<tr>
<td>ILMN_Gene</td>
<td>Illumina Gene</td>
<td>Illumina Gene</td>
</tr>
<tr>
<td>Protein Product</td>
<td>Protein product</td>
<td>Protein product</td>
</tr>
<tr>
<td>protein_domains</td>
<td>protein_domains</td>
<td>protein_domains</td>
</tr>
<tr>
<td>Array Address Id</td>
<td>Array adress id</td>
<td>Array adress id</td>
</tr>
<tr>
<td>Probe_Sequence sequencname</td>
<td>Probe coordinates</td>
<td>Probe coordinates</td>
</tr>
<tr>
<td>Chromosome</td>
<td>Chromosome</td>
<td>Chromosome</td>
</tr>
<tr>
<td>strand</td>
<td>Strand</td>
<td>Strand</td>
</tr>
<tr>
<td>Probe Chr Orientation</td>
<td>Probe chr orientation</td>
<td>Probe chr orientation</td>
</tr>
<tr>
<td>Probe Coordinates</td>
<td>Probe coordinates</td>
<td>Probe coordinates</td>
</tr>
<tr>
<td>Obsolete Probe id</td>
<td>Obsolete probe id</td>
<td>Obsolete probe id</td>
</tr>
</tbody>
</table>
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.


