Biomedical Genomics Workbench

REFERENCE MANUAL

Manual for

Biomedical Genomics Workbench 5.0.1

Windows, macOS and Linux

June 1, 2018

This software is for research purposes only.

QIAGEN Aarhus
Silkeborgvej 2
Prismet
DK-8000 Aarhus C
Denmark

Contents

Part I

Introduction

Chapter 1

Introduction to *Biomedical Genomics Workbench*

Contents

Welcome to *Biomedical Genomics Workbench 5.0.1* — a software package supporting your daily bioinformatics work.

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

This software is for research purposes only.

1.1 Contact information

The *Biomedical Genomics Workbench* is developed by:

QIAGEN Aarhus
Silkeborgvej 2
Prismet
8000 Aarhus C
Denmark

http://www.qiagenbioinformatics.com

Email: ts-bioinformatics@qiagen.com

The QIAGEN Aarhus team is continuously improving the *Biomedical Genomics Workbench* with our users' interests in mind. We welcome all requests and feedback from users, as well as suggestions for new features or more general improvements to the program.

Getting help via the workbench

If you encounter a problem or need help understanding how the *Biomedical Genomics Workbench* works, and the license you are using is covered by our Maintenance, Upgrades and Support (MUS) program (http://www.qiagenbioinformatics.com/maintenance-and-support/), you can contact our customer support via the workbench by going to the menu option:

Help | **Contact Support**

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

Other ways to contact the support team

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

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

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

1.2 Download and installation

The *Biomedical Genomics Workbench* is developed for Windows, macOS and Linux. The software for either platform can be downloaded from http://www.qiagenbioinformatics.com/

product-downloads/.

1.2.1 Program download

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

In the dialog you must choose:

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

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

1.2.2 Installation on Microsoft Windows

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

Installing the program is done in the following steps:

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

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

1.2.3 Installation on macOS

Starting the installation process is done in the following way: When you have downloaded an installer, locate the downloaded installer and double-click the icon. The default location for downloaded files is your desktop.

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

Installing the program is done in the following steps:

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

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

1.2.4 Installation on Linux with an installer

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

```
# sh BiomedicalGenomicsWorkbench_5_0_64
```

Installing the program is done in the following steps:

- On the welcome screen, click **Next**.
- Read and accept the License agreement and click **Next**.
- Choose where you would like to install the application and click **Next**.
  For a system-wide installation you can choose for example /opt or /usr/local. If you do not have root privileges you can choose to install in your home directory.

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

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

```
# biomedicalgenomicswb5
```

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

```
# ./biomedicalgenomicswb5
```

1.3 System requirements

- Windows 7, Windows 8, Windows 10, Windows Server 2012, and Windows Server 2016
- OS X 10.10, 10.11 and macOS 10.12, 10.13
- Linux: RHEL 6.7 and later, SUSE Linux Enterprise Server 11 and later. The software is expected to run without problem on other recent Linux systems, but we do not guarantee this.
- 64 bit operating system
- 16 GB RAM required
- 24 GB RAM recommended
- 1024 x 768 display required
- 1600 x 1200 display recommended
- Intel or AMD CPU required
- Minimum 100 GB free disk space in the tmp directory
- Minimum 90 GB free disk space required in the CLC_References directory if you are not connected to a server and wish to work with either hg19 or hg38. For more information about reference data size, see section **??**. If you have less free disk space available it is possible to change the reference data location. How to do this is described in section 13.1.
- **Special requirements for the 3D Molecule Viewer**
  - **System requirements**
    - A graphics card capable of supporting OpenGL 2.0.

    - Updated graphics drivers. Please make sure the latest driver for the graphics card is installed.

  - **System Recommendations**

    - A discrete graphics card from either Nvidia or AMD/ATI. Modern integrated graphics cards (such as the Intel HD Graphics series) may also be used, but these are usually slower than the discrete cards.

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

- **Special requirements for read mapping**. The numbers below give minimum and recommended memory for systems running mapping and analysis tasks based on the genome size.

  - **Human (3.2 gigabases)** and **Mouse (2.7 gigabases)**

    - Minimum: 6 GB RAM
    - Recommended: 8 GB RAM

1.3.1 Limitations on maximum number of cores

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

For static licenses, there is a limitation on the number of logical cores on the computer. If there are more than 64 logical cores, the *Biomedical Genomics Workbench* cannot be started. In this case, a network license is needed (read more at http://www.qiagenbioinformatics.com/support/licensing/).

1.4 Workbench Licenses

When you have installed the workbench and start it for the first time, or after installing a new major release, you will meet the license assistant, shown in figure 1.1.

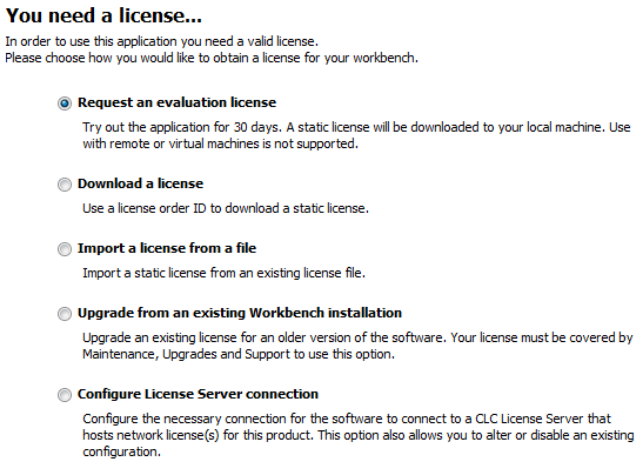


Figure 1.1: *The license assistant showing you the options for getting started.*

To install a license, you must be running the program in administrative mode. 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".

The **License Manager** can also be accessed from the menu bar in the Workbench:

Help | **License Manager**

The following options are available. They are described 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 received 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 web-based licensing system.
- **Upgrade license**. If you have used a previous version of the *Biomedical Genomics Workbench*, and you are entitled to upgrade to a new major version, select this option to upgrade your license file.
- **Configure license server connection**. If your organization has a CLC License Server, select this option to configure the connection to it.

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

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

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

1.4.1 Request an evaluation license

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

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

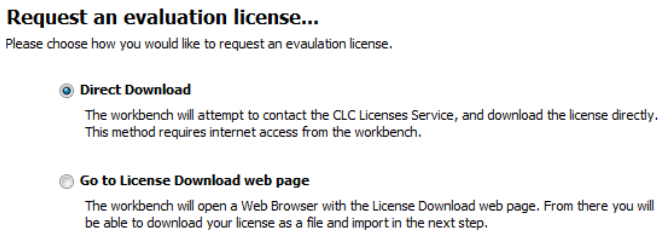


Figure 1.2: *Choosing between direct download or going to the license download web page.*

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 license download web page**. In a browser window, show the license download web page, which can be used to download a license file. This option is suitable in situations where, for example, you are working behind a proxy, so that the Workbench does not have direct access to the CLC Licenses Service.

If you select the option to download a license directly and it turns out that the Workbench does not have direct access to the external network, (because of a firewall, proxy server etc.), you can click **Previous** button to try the other method.

After selection on 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**, the dialog shown in figure 1.3 appears.

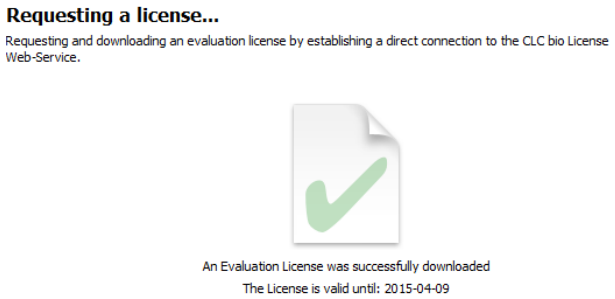


Figure 1.3: *A license has been downloaded.*

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

Go to license download web page

After choosing the *Go to license download web page* option and clicking on the button labeled **Next**, the license download web page appears in a browser window, as shown in 1.4.

Click the **Request Evaluation License** button. You can then save the license on your system.

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

Click the **Choose License File** button and browse to find the license file you saved. When you have selected the file, click on the button labeled **Next**.

Accepting the license agreement

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 1.25.

Figure 1.4: *The license download web page.*

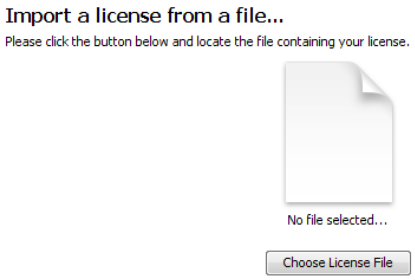


Figure 1.5: *Importing the license file downloaded from the web page.*

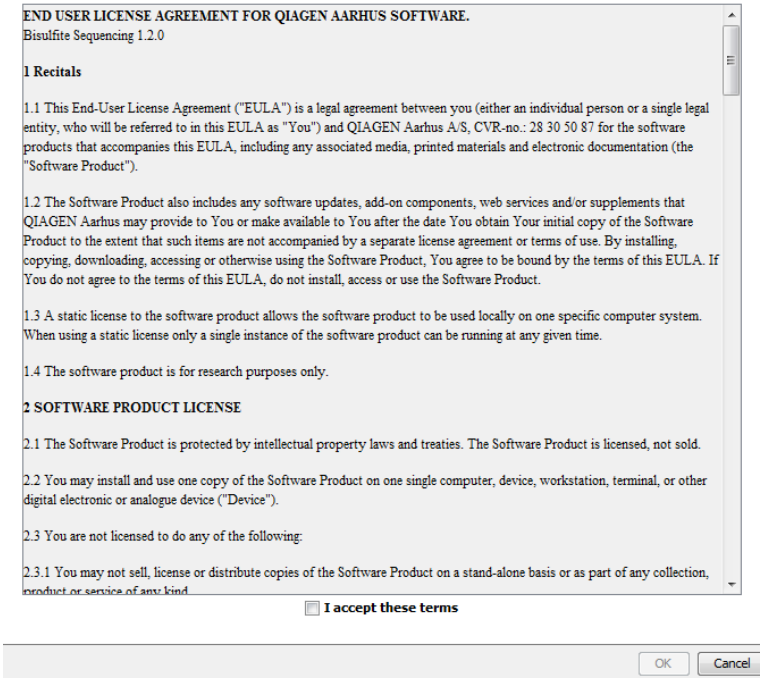


Figure 1.6: *Read the license agreement carefully.*

Please read the EULA text carefully before clicking in the box next to the text **I accept these terms** to accept. If requested, fill in your personal information before clicking **Finish**.

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 **Next** button, you will see the dialog shown in 1.7. Enter your license order ID into the text field under the title License Order-ID. (The ID can be pasted into the box after copying it and then using menus or key combinations like Ctrl+V on some system or ⌘ + V on Mac).

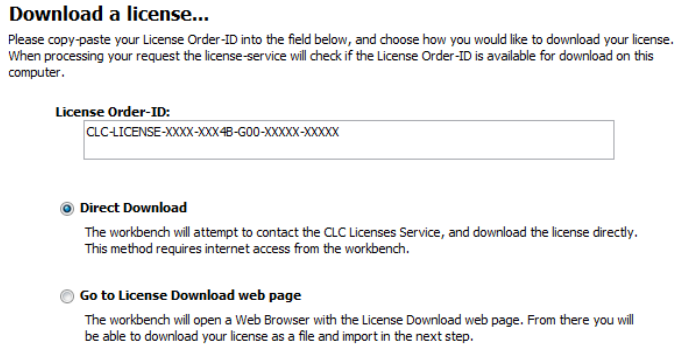


Figure 1.7: *Enter a license order ID for the software.*

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 license download web page**. In a browser window, show the license download web page, which can be used to download a license file. This option is suitable in situations where, for example, you are working behind a proxy, so that the Workbench does not have direct access to the CLC Licenses Service.

If you select the option to download a license directly and it turns out that the Workbench does not have direct access to the external network, (because of a firewall, proxy server etc.), you can click **Previous** button to try the other method.

After selection on 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**, the dialog shown in figure 1.8 appears.

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

Go to license download web page

After choosing the *Go to license download web page* option and clicking on the button labeled **Next**, the license download web page appears in a browser window, as shown in 1.9.

Click the **Request Evaluation License** button. You can then save the license on your system.

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

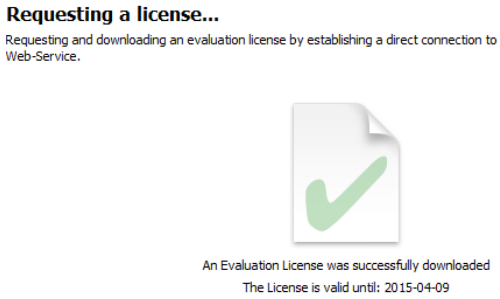


Figure 1.8: *A license has been downloaded.*

Figure 1.9: *The license download web page.*

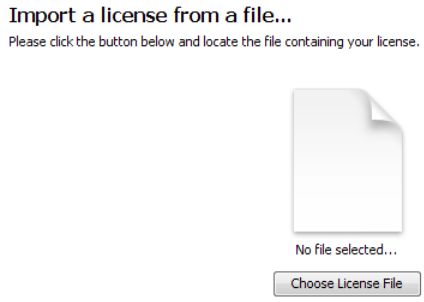


Figure 1.10: *Importing the license file downloaded from the web page.*

Click the **Choose License File** button and browse to find the license file you saved. When you have selected the file, click on the button labeled **Next**.

Accepting the license agreement

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 1.25.

Please read the EULA text carefully before clicking in the box next to the text **I accept these terms** to accept. If requested, fill in your personal information before clicking **Finish**.

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.

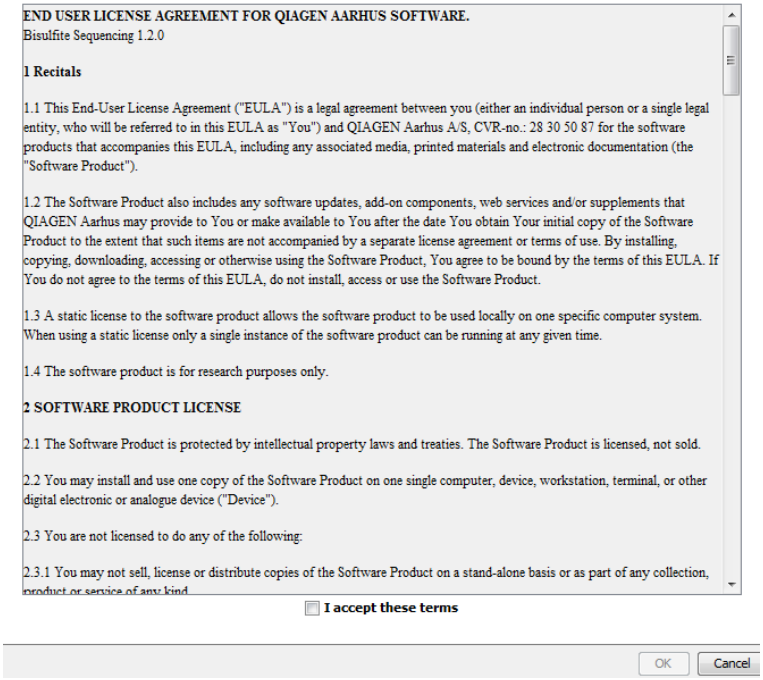


Figure 1.11: *Read the license agreement carefully.*

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

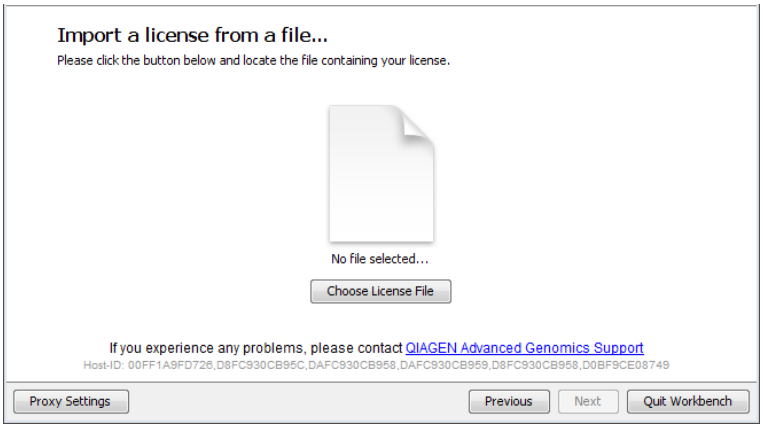


Figure 1.12: *Selecting a license file.*

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

Accepting the license agreement

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 1.25.

Please read the EULA text carefully before clicking in the box next to the text **I accept these terms** to accept. If requested, fill in your personal information before clicking **Finish**.

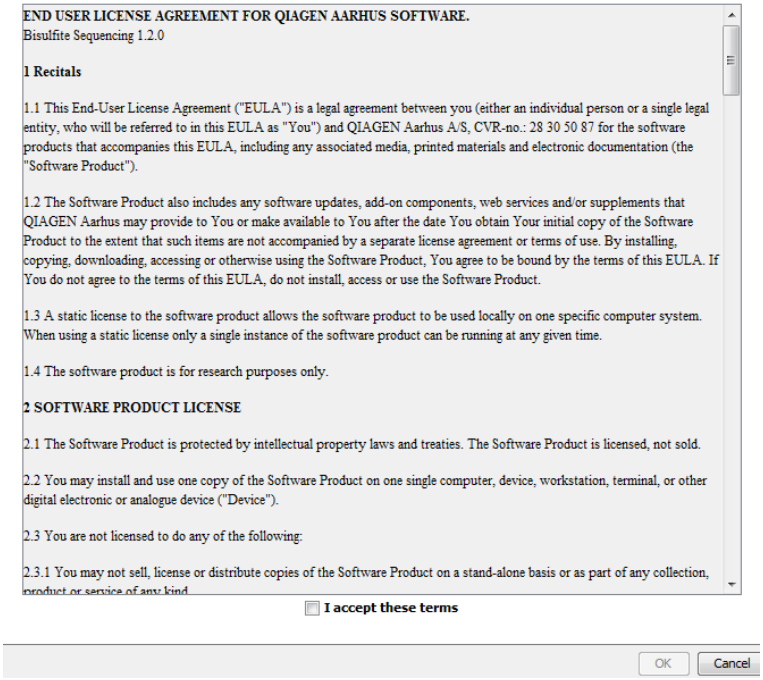


Figure 1.13: *Read the license agreement carefully.*

1.4.4 Upgrade license

This option is used when you already have used a previous version of *Biomedical Genomics Workbench*, and you are entitled to upgrade to a new major version. The Workbench will need direct access to the external network to use this option.

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

If the Workbench finds an existing license file, the next dialog will look like figure 1.14.

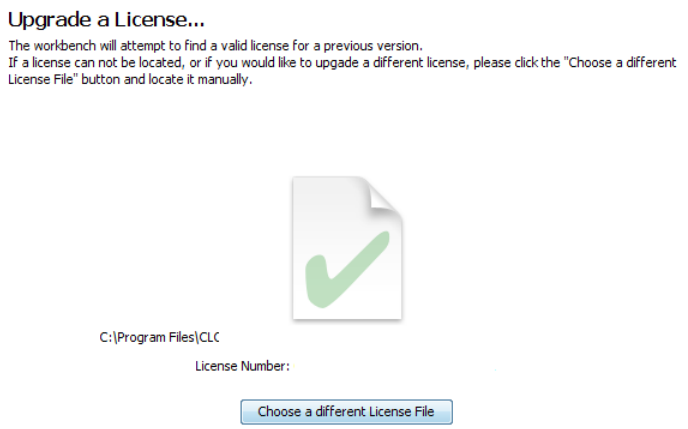


Figure 1.14: *An license from an older installation is found.*

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

If the Workbench cannot connect to the external network directly, please see the section on downloading a license for non-networked machines. You will need your license order ID for this.

Your license must be covered by our Maintenance, Upgrades and Support (MUS) program to be eligible to upgrade your license. If the license is covered for upgrades and there are any problems with this, please contact bioinformaticslicense@qiagen.com.

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 license download web page**. In a browser window, show the license download web page, which can be used to download a license file. This option is suitable in situations where, for example, you are working behind a proxy, so that the Workbench does not have direct access to the CLC Licenses Service.

If you select the option to download a license directly and it turns out that the Workbench does not have direct access to the external network, (because of a firewall, proxy server etc.), you can click **Previous** button to try the other method.

After selection on 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**, the dialog shown in figure 1.15 appears.

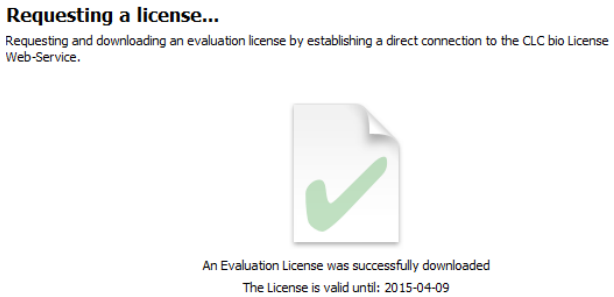


Figure 1.15: *A license has been downloaded.*

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

Go to license download web page

After choosing the *Go to license download web page* option and clicking on the button labeled **Next**, the license download web page appears in a browser window, as shown in 1.16.

Click the **Request Evaluation License** button. You can then save the license on your system.

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

Click the **Choose License File** button and browse to find the license file you saved. When you have selected the file, click on the button labeled **Next**.

Figure 1.16: *The license download web page.*

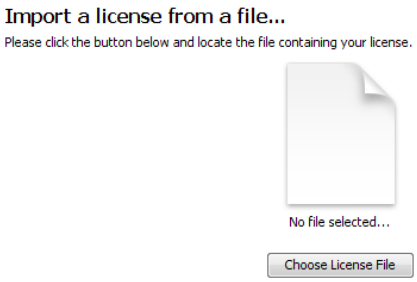


Figure 1.17: *Importing the license file downloaded from the web page.*

Accepting the license agreement

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 1.25.

Please read the EULA text carefully before clicking in the box next to the text **I accept these terms** to accept. If requested, fill in your personal information before clicking **Finish**.

1.4.5 Configure license server connection

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

To configure the Workbench to connect to a CLC License Server, select the **Configure License Server connection** option and click on the **Next** button. A dialog for the license server connection configuration is then presented. See figure 1.19.

The options in that dialog are:

- **Enable license server connection**. This box must be checked for the Workbench is to contact the CLC License Server to get a license for *Biomedical Genomics Workbench*.
- **Automatically detect license server**. By checking this option the Workbench will look for a CLC License Server accessible from the Workbench. Automatic server discovery sends UDP broadcasts from the Workbench on port 6200. Available license servers respond to the broadcast. The Workbench then uses TCP communication for to get a license, if one is available. Automatic server discovery works only on local networks and will not work

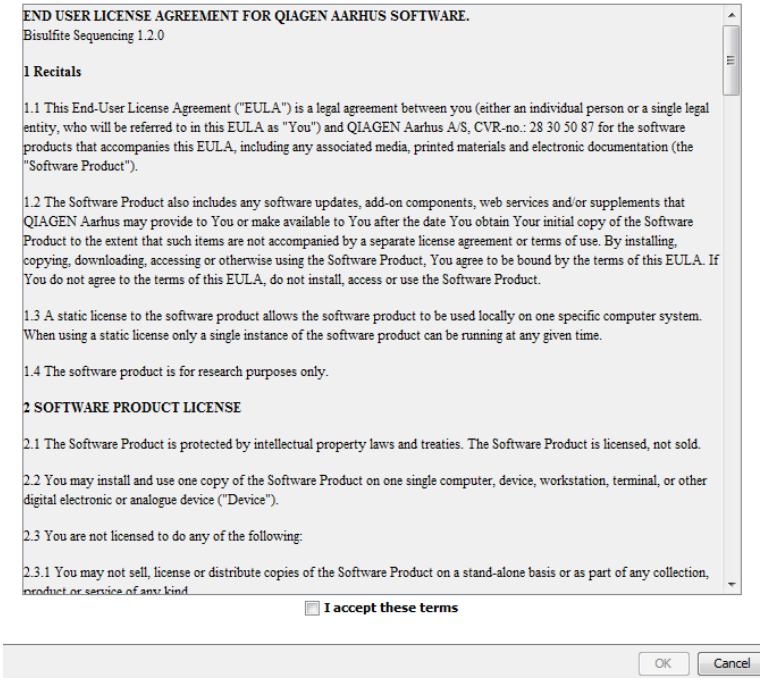


Figure 1.18: *Read the license agreement carefully.*

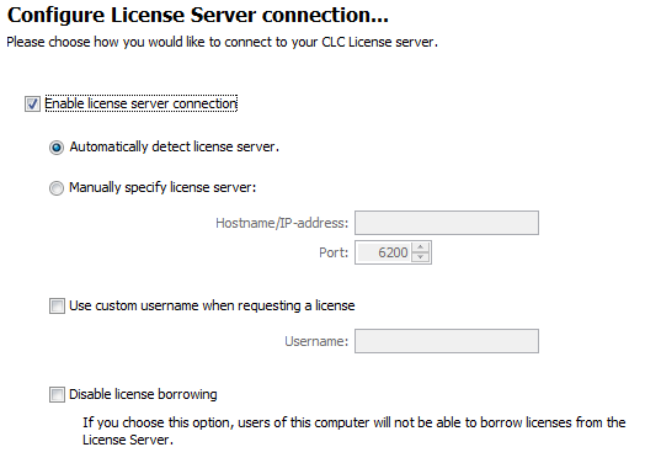


Figure 1.19: *Connecting to a CLC License Server.*

on WAN or VPN connections. Automatic server discovery is not guaranteed to work on all networks. If you are working on an enterprise network on where local firewalls or routers cut off UDP broadcast traffic, then you may need to configure the details of the CLC License Server using the **Manually specify license server** option instead.

- **Manually specify license server**. Select this option to enter the details of the machine the CLC License Server software is running on, specifically:
  - **Host name**. The address for the machine the CLC Licenser Server software is running on.
  - **Port**. The port used by the CLC License Server to receive requests.
- **Use custom username when requesting a license**. Optional. If this is checked, a username can be entered. That will be passed to the CLC License Server instead of the username of

the account being used to run the Workbench.

- **Disable license borrowing on this computer**. Check this box if you do not want users of the computer to borrow a license. See section 1.4.5 for further details.

Special note on modules needing a license

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

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

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

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

Borrowing a license

A network license can only be used when the Workbench is connected to the license server. If you wish to use the *Biomedical Genomics Workbench* when you are not connected to the CLC License Server, you can *borrow* an available license for a period of time. During this time, there will be one less network license available for other users. The Workbench must have a connection to the CLC License Server at the point in time when you wish to borrow a license.

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 1.20.

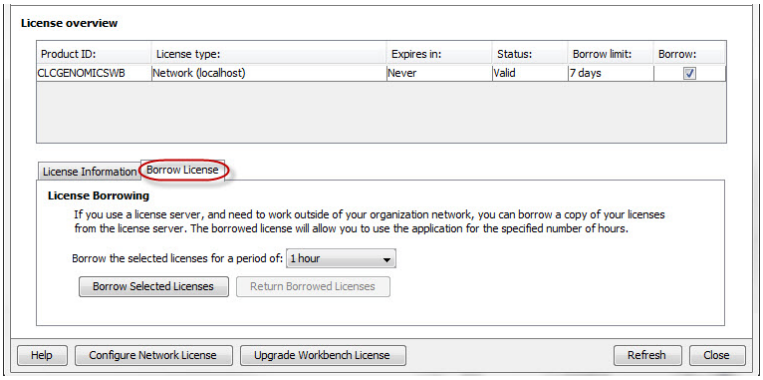


Figure 1.20: *Borrow a license.*

3. Use the checkboxes at the right hand side of the table in the License overview section of the window to select the license(s) that you wish to borrow.
4. Select the length of time you wish to borrow the license(s).
5. Click on the button labeled **Borrow Licenses**.
6. Close the License Manager when you are done.

You can now go offline and work with *Biomedical Genomics Workbench*. When the time period you borrowed the license for has elapsed, the network license you borrowed is made available again for other users to access. To continue using *Biomedical Genomics Workbench* with a license, you will need to connect the Workbench to the network again so it can contact the CLC Licence Server to obtain one.

Note! Your CLC License Server administrator can choose to disable to the option allowing the borrowing of licenses. If this has been done, you will not be able to borrow a network license using your Workbench.

Common issues when using a network license

No license available at the moment If all the licenses are in use, you will see a dialog like that shown in figure 1.21 when you start up the Workbench.

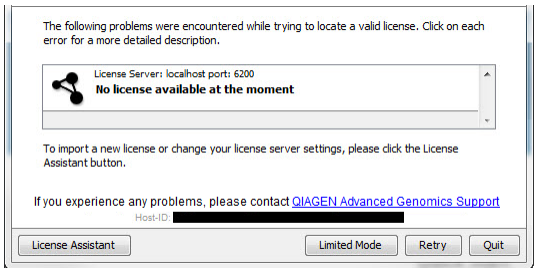


Figure 1.21: *This window appears when there are no available network licenses for the software you are running.*

This means others are using the network licenses. You will need to wait for them to return their licenses before you can continue to work with a fully functional copy of the software. If this is a frequent issue, you may wish to discuss this with your CLC License Server administrator.

Clicking on the **Viewing Mode** button in the dialog allows you to start *Biomedical Genomics Workbench* for data import, export, the ability to access your CLC data and to run a few selected tools.

Lost connection to the CLC License Server If the Workbench connection to the CLC License Server is lost, you will see a dialog as shown in figure 1.22.

If you have chosen the option to **Automatically detect license server** and you have not succeeded in connecting to the 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 possible at your site, you will need to manually configure the CLC License Server settings using the License Manager, as described earlier in this section.

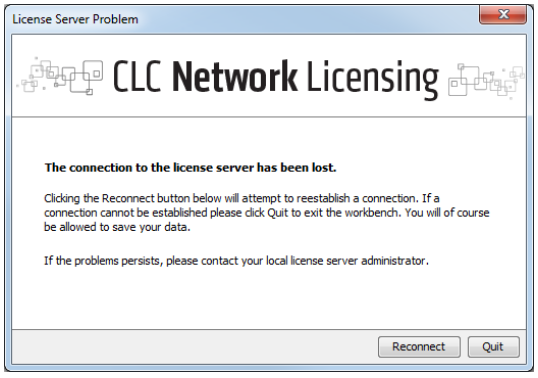


Figure 1.22: *This message appears if the Workbench is unable to establish a connection to a CLC License server.*

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

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

Help | License Manager ( )

The license manager is shown in figure 1.23.

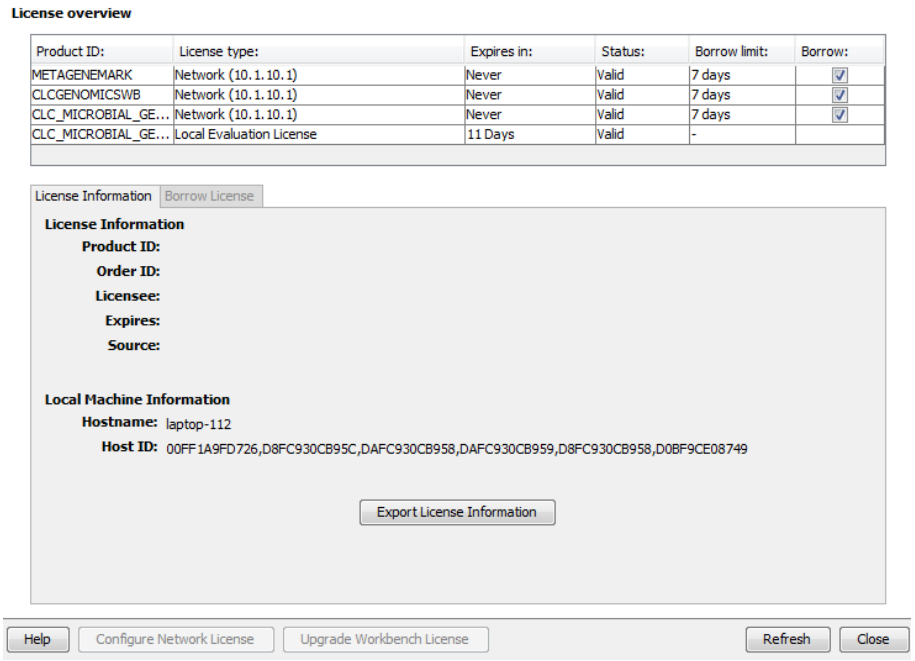


Figure 1.23: *The license manager.*

This dialog can be used to:

- See information about the license (e.g., what kind of license, when it expires)

- Configure how to connect to a license server using the **Configure Network License** button at the lower left corner to open the dialog seen in figure 1.19.
- Upgrade from an evaluation license by clicking the **Upgrade Workbench License** button to open the dialog shown in figure 1.1.
- Export license information to a text file.
- Borrow a 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 *Biomedical Genomics Workbench* on the machine you wish to run the software on.
- Start up the software as an administrative user and find the host ID of the machine that you will run the CLC Workbench on. You can see the host ID the machine reported at the bottom of the License Manager window in grey text.
- Make a copy of this host ID such that you can use it on a machine that has internet access.
- Go to a computer with internet access, open a browser window and go to the relevant network license download web page:
- For workbenches released from January 2013 and later, (e.g. the Genomics Workbench version 6.0 or higher, and the Main Workbench, version 6.8 or higher), please go to:

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

  For earlier workbenches, please go to:

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

  It is crucial that you choose the license download page appropriate to the version of the software you plan to run.
- Paste in your license order ID and the host ID that you noted down in the relevant boxes on the webpage.
- Click 'download license' and save the resulting .lic file.
- Open the Workbench on your non-networked machine. In the Workbench license manager choose 'Import a license from a file'. In the resulting dialog click 'choose license file' to browse the location of the .lic file you have just downloaded.

  If the License Manager does not start up by default, you can start it up by going to the Help menu and choosing License Manager.
- Click on the **Next** button and go through the remaining steps of the license manager wizard.

1.4.7 Viewing mode

CLC Workbenches without a valid license can be run in Viewing Mode. This mode allows you to import, export and view data stored in the Workbench or a CLC Server.

Certain data types require viewing functionality provided by plugins or modules, and these can be installed when running in Viewing Mode.

Some basic analysis tools are available when in Viewing Mode. Those can be found under the Toolbox menu.

To go from running in Viewing Mode to running a Workbench with its full functionality, restart the Workbench such that it has access to a valid license.

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

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 *Biomedical Genomics Workbench* again (without pressing Shift).

1.5 When the program is installed: Getting started

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

Tutorials describing hands-on examples of how to use the individual tools and features of the *Biomedical Genomics Workbench* can be found at http://www.qiagenbioinformatics.com/support/tutorials/. We also recommend our **Online presentations** where a product specialist demonstrates our software. This is a very easy way to get started using the program. Read more about video tutorials and other online presentations here: http://tv.qiagenbioinformatics.com/.

1.6 Plugins

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

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

Note: In order to install plugins and modules, the Workbench must be run in administrator mode. On Linux and Mac, it means you must be logged in as an administrator. On Windows, you can do

this by right-clicking the program shortcut and choosing "Run as Administrator".

Plugins are installed and uninstalled using the plugin manager.

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

The plugin manager has two tabs at the top:

- **Manage Plugins.** This is an overview of plugins that are installed.
- **Download Plugins.** This is an overview of available plugins on QIAGEN Aarhus server.

1.6.1 Install

To install a plugin, click the **Download Plugins** tab. This will display an overview of the plugins that are available for download and installation (see figure 1.24).

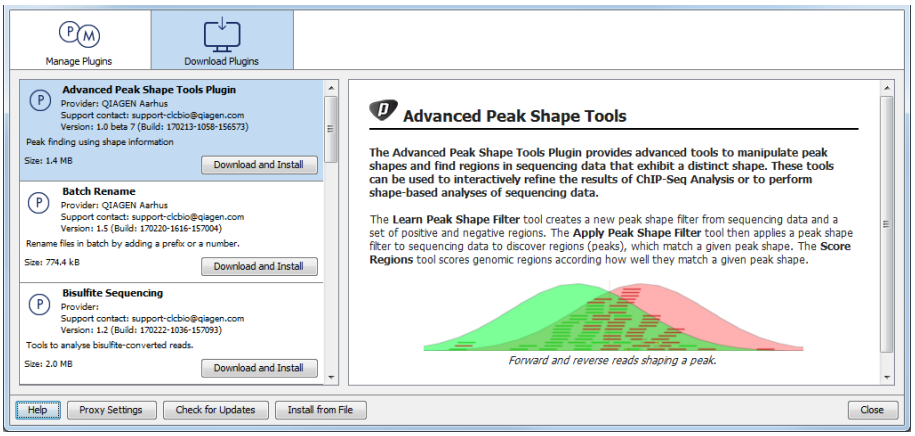


Figure 1.24: *The plugins that are available for download.*

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

Accepting the license agreement

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 1.25.

Please read the EULA text carefully before clicking in the box next to the text **I accept these terms** to accept. If requested, fill in your personal information before clicking **Finish**.

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

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

1.6.2 Uninstall

Plugins are uninstalled using the plugin manager:

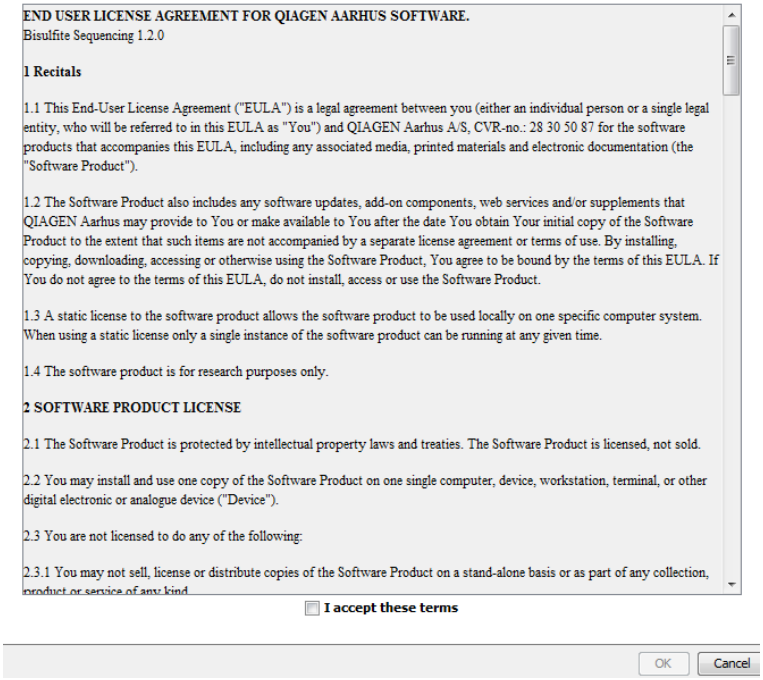


Figure 1.25: *Read the license agreement carefully.*

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

This will open the dialog shown in figure 1.26.

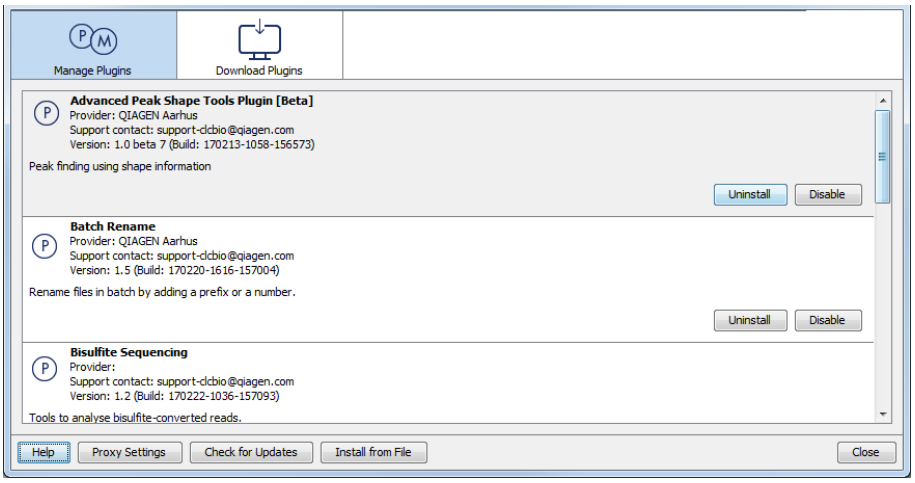


Figure 1.26: *The plugin manager with plugins installed.*

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

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

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

1.6.3 Updating plugins

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

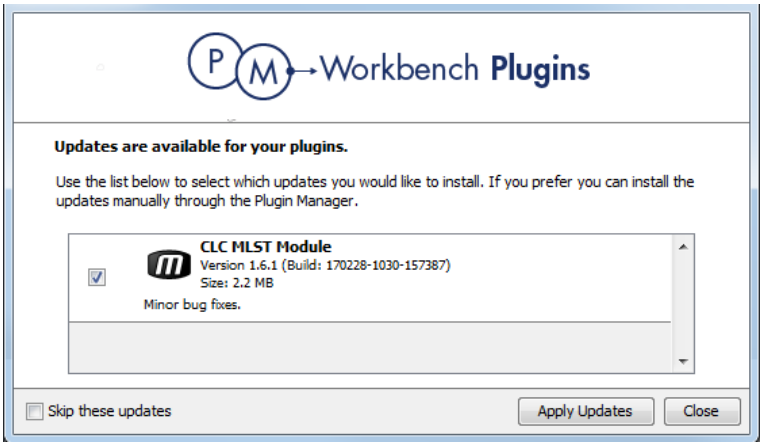


Figure 1.27: *Plugin updates.*

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 1.26).

1.7 Network configuration

If you use a proxy server to access the Internet you must configure *Biomedical Genomics Workbench* to use this. Otherwise you will not be able to perform any online activities.

Biomedical Genomics Workbench supports the use of an HTTP-proxy and an anonymous SOCKS-proxy.

To configure your proxy settings, open the workbench, go to **Edit | Preferences** and choose the **Advanced** tab (figure 1.28).

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

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

Exclude hosts can be used if there are some hosts that should be contacted directly and not through the proxy server. The value can be a list of hosts, each separated by a `|`, and in addition a wildcard character `*` can be used for matching. For example: `*.foo.com|localhost`.

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

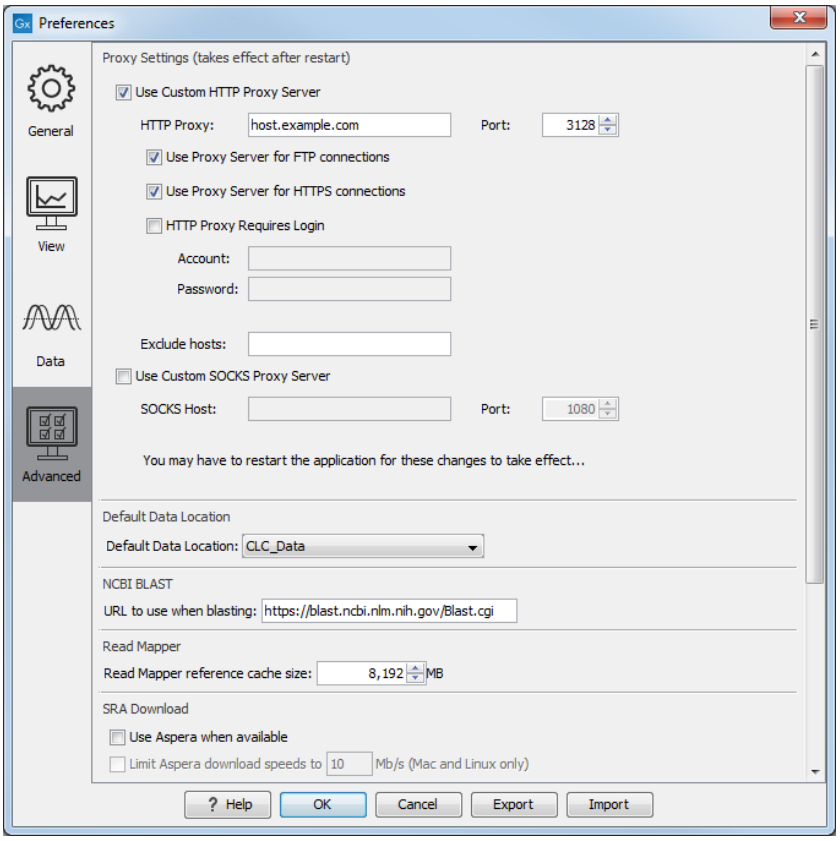


Figure 1.28: *Adjusting proxy preferences.*

Part II

Core functionalities

Chapter 2

User interface

Contents

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

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

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

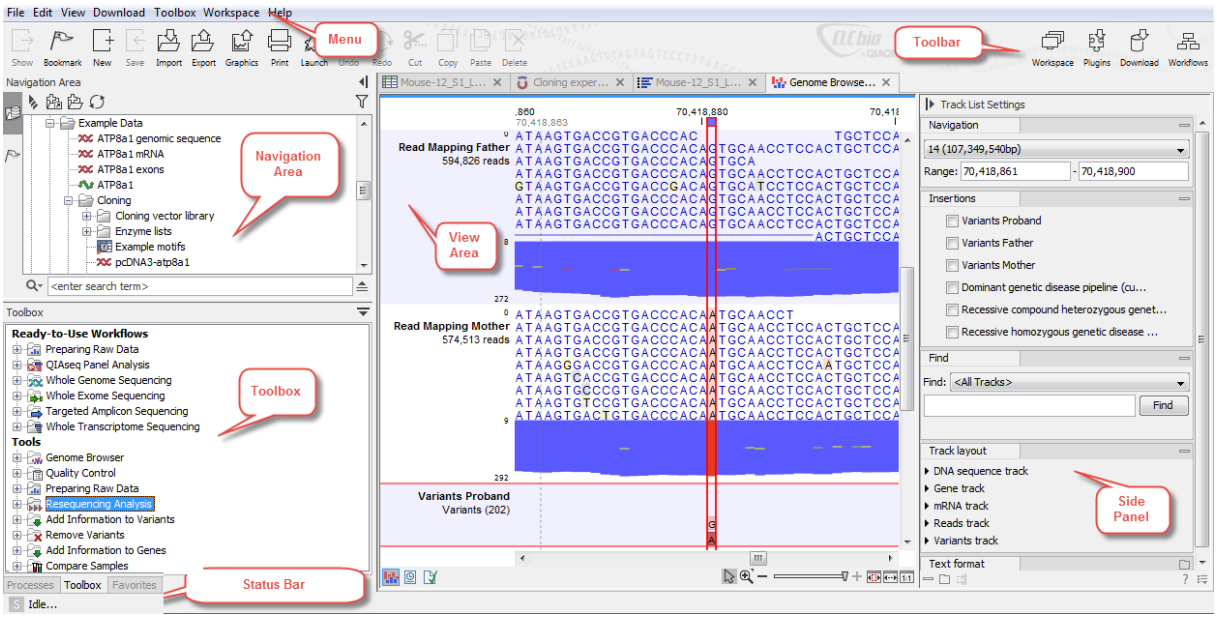


Figure 2.1: *The user interface.*

2.1 View Area

The **View Area** is the central part of the screen, displaying your current work. The View Area may consist of one or more **Views**, represented by **tabs** at the top of the View Area. In figure 2.2, four views are displayed: three as tabs in the upper view, and one in an horizontal split view. The tab currently selected, i.e., active, is indicated by a blue bar underneath the tab (here the bottom tab open in the bottom view).

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

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

2.1.1 Open view

Elements

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

double-click an element in the Navigation Area

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

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

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

clicking its tab.

Views

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

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

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

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

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

Split views

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

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

This will open a split view with a linear view at the bottom and a circular view at the top (see 10.8).

You can also show a circular view of a sequence without opening the sequence first:

Select the sequence in the Navigation Area | Show () | As Circular ()

2.1.2 History and Info views

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

The History view is a textual log of all operations you make in the program. If for example you rename a sequence, align sequences, create a phylogenetic tree or translate a sequence, you can always go back and check what you have done. In this way, you are able to document and reproduce previous operations.

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

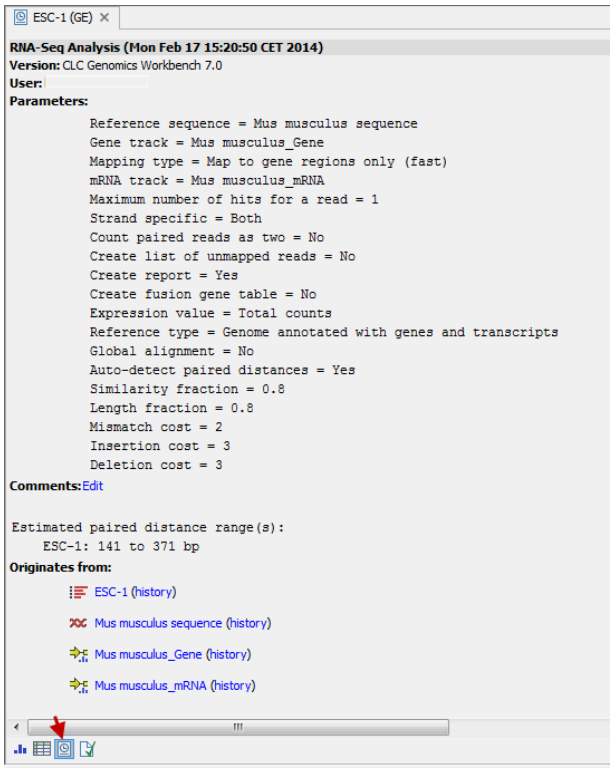


Figure 2.4: *An element's history.*

The following information is available:

- **Originates from workflow** (optional). In cases where the file was generated by a workflow, the first line will state the Name and Version number of that workflow.
- **Title**. The action that the user performed.
- **Date and time**. Date and time for the operation. The date and time are displayed according to your locale settings (see section 4.1).
- **Version**. The workbench type and version that has been used.
- **User**. The user who performed the operation. If you import some data created by another person in a CLC Workbench, that person's name will be shown.
- **Parameters**. Details about the action performed. This could be the parameters that were chosen for an analysis.
- **Comments**. By clicking **Edit** you can enter your own comments regarding this entry in the history. These comments are saved.
- **Originates from**. This information is usually shown at the bottom of an element's history. Here, you can see which elements the current element originates from. For example, if you have created an alignment of three sequences, the three sequences are shown here. Clicking the element selects it in the Navigation Area, and clicking the "history" link opens the element's own history.

When an element's info is open you can check current information about the element, and in particular the potential association of the data you are looking at with metadata. To learn more about the **Show Element Info** button, see section 10.4 and see section 3.2.4.

2.1.3 Close views

When a view is closed, the **View Area** remains open as long as there is at least one open view.

A view is closed by:

Right-click the tab | **Close** or **Select the view** | **Ctrl + W**

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

- **Close.** See above.
- **Close Other Tabs.** Closes all other tabs, in all tab areas, except the one that is selected.
- **Close Tab Area.** Closes all tabs in the tab area, but not the tabs that are in split view.
- **Close All Tabs.** Closes all tabs, in all tab areas. Leaves an empty workspace.

2.1.4 Save changes in a view

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

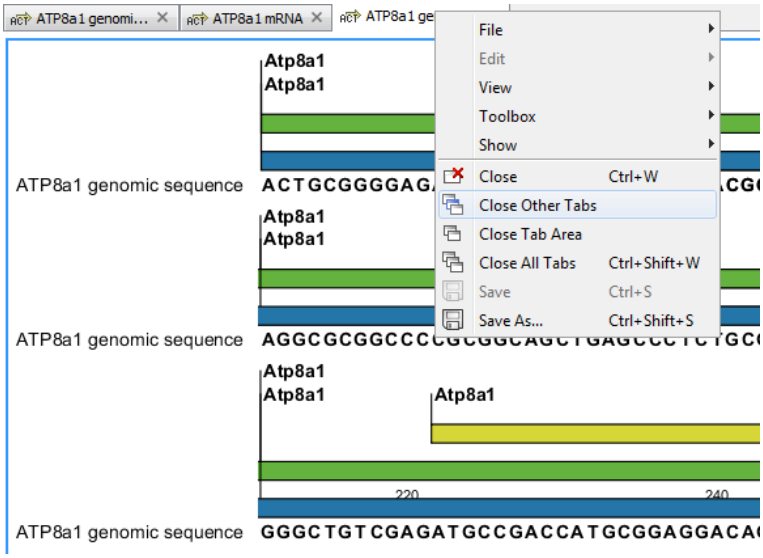


Figure 2.5: *By right-clicking a tab, several close options are available.*

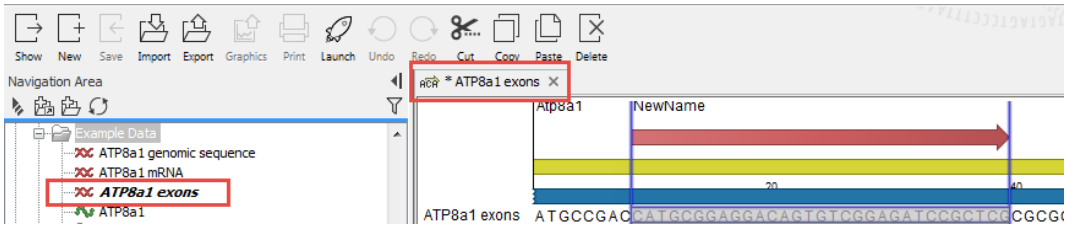


Figure 2.6: *An * on a tab name always indicates that the view is unsaved. In this case, an existing element was edited but not saved yet, so the element's name is also highlighted in bold and italic in the Navigation Area.*

The **Save** function may be activated in two ways: Select the tab of the view you want to save and

Save (💾) or Ctrl + S (⌘ + S on Mac)

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

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

2.1.5 Undo/Redo

If you make a change to an element in a view, e.g. remove an annotation in a sequence or modify a tree, you can undo the action. In general, **Undo** applies to all changes you can make when right-clicking in a view. **Undo** is done by:

Click undo (↶) in the Toolbar or **Ctrl + Z**

If you want to undo several actions, just repeat the steps above.

To reverse the undo action:

Click the redo icon in the Toolbar or **Ctrl + Y**

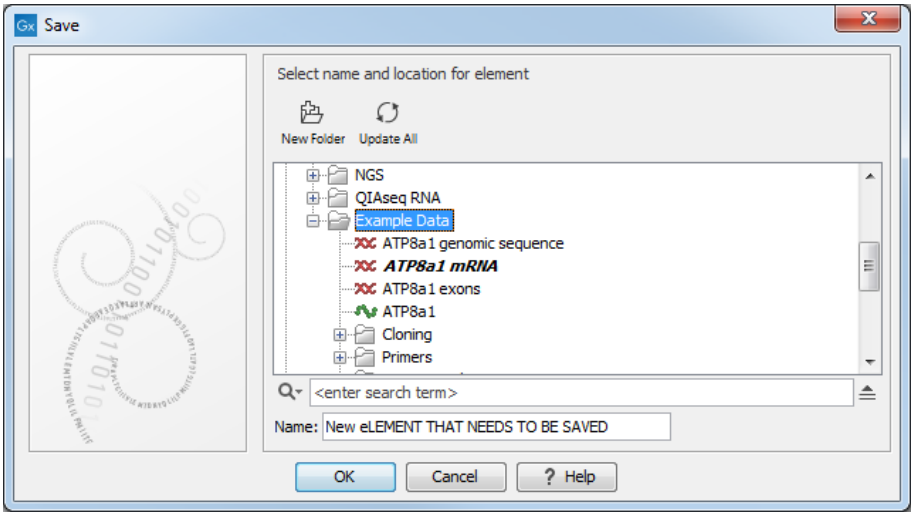


Figure 2.7: *Save dialog. The new element has been name "New element that needs to be saved" and will be saved in the "Example Data" folder.*

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

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

2.1.6 Arrange views in View Area

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

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

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

You can also split a View Area horizontally or vertically using the menus.

Splitting horizontally may be done this way:

right-click a tab of the view | **View** | **Split Horizontally ( )**

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

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

Select the view you want to maximize, and click

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

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

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

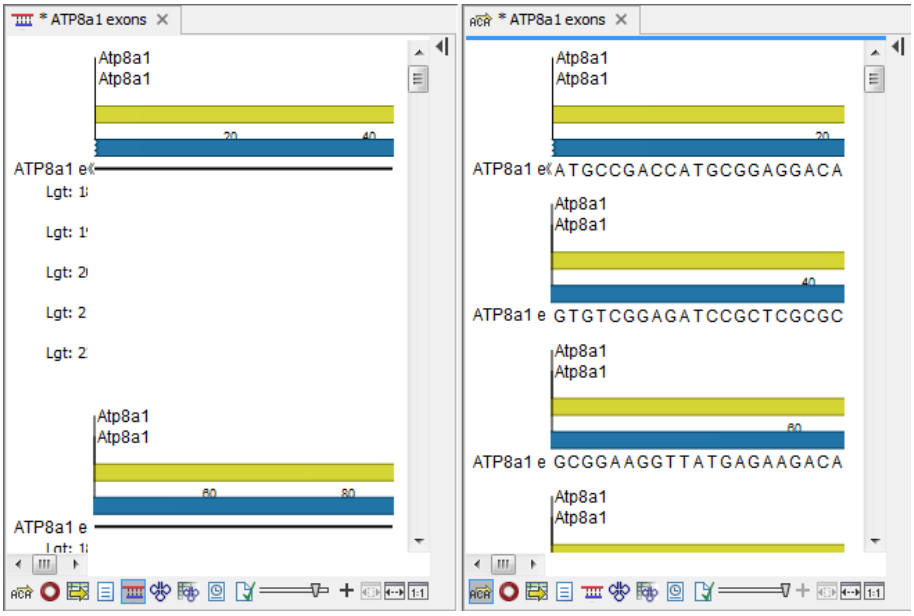


Figure 2.8: *A vertical split screen.*

The following restores the size of the view:

View | **Maximize/restore View (**▭**)** or **Ctrl + M**

or **double-click title of view**

2.1.7 Moving a view to a different screen

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

An example is shown in figure 2.9, where the main Workbench window shows a table of open reading frames, and the screen to the right is used to display the sequence and annotations.

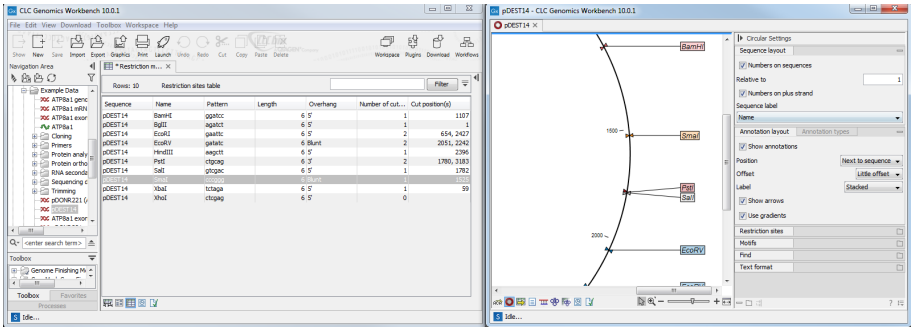


Figure 2.9: *Showing the table on one screen while the sequence is displayed on another screen. Clicking the table of open reading frames causes the view on the other screen to follow the selection.*

You can make more detached windows, by dropping tabs outside the open workbench windows, or you can drag more tabs to a detached window. To get a tab back to the main workbench

window, just drag the detached tab back, and drop it next to the other tabs in the top of the view area. **Note:** You should not drag the detached window header, just the tab itself.

2.1.8 Side Panel

The **Side Panel** allows you to change the way the content of a view is displayed. The options in the Side Panel depend on the kind of data in the view, and they are described in the relevant sections about sequences, alignments, trees etc.

Figure 2.10 shows the default Side Panel for a protein sequence. It is organized into **palettes**.

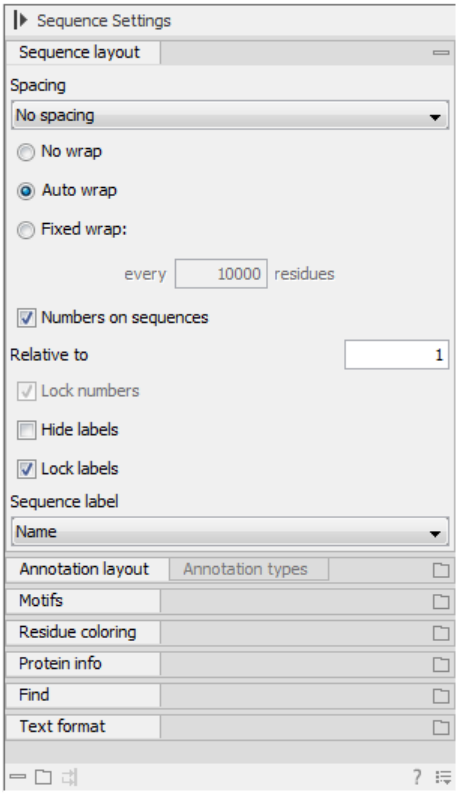


Figure 2.10: *The default view of the Side Panel when opening a protein sequence.*

In this example, there is one palette for Sequence layout, one for Annotation Layout etc. These palettes can be re-organized by dragging the palette name with the mouse and dropping it where you want it to be. They can either be situated next to each other, so that you can switch between them, or they can be listed on top of each other, so that expanding one of the palettes will push the palettes below further down.

In addition, they can be moved away from the Side Panel and placed anywhere on the screen as shown in figure 2.11.

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

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

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

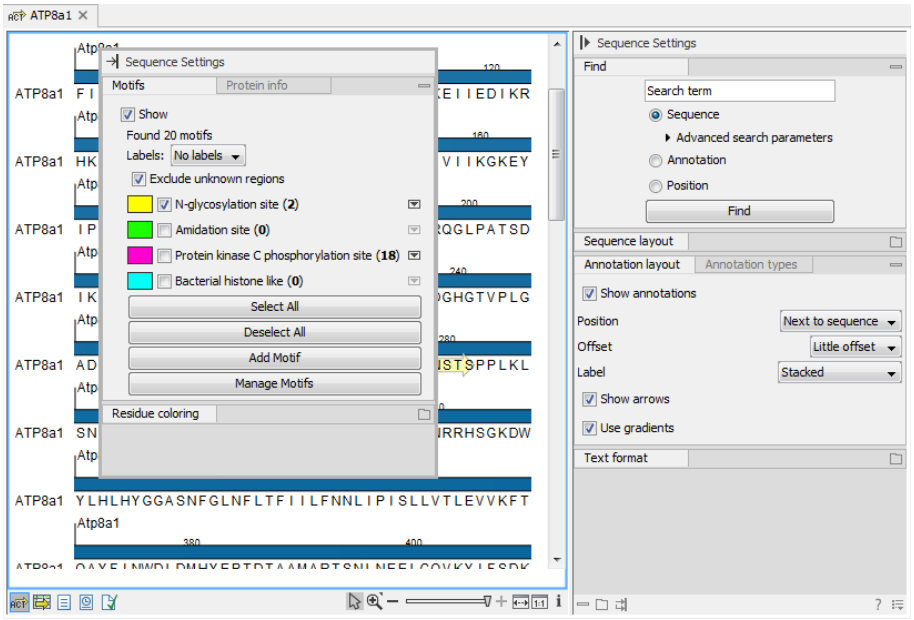


Figure 2.11: *Palettes can be organized in the Side Panel as you like or placed anywhere on the screen.*

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

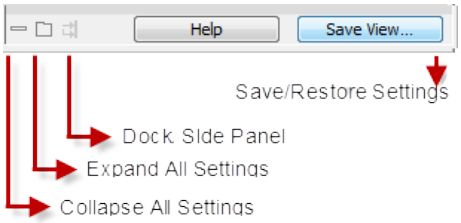


Figure 2.12: *Functionalities found at the bottom of the Side Panel.*

- Collapse all settings ( ).
- Expand all settings ( ).
- Dock all palettes ( )
- Get **Help** for the particular view and settings
- Save the settings of the Side Panel or apply already saved settings. Changes made to the Side Panel, including the organization of palettes, will not be saved when you save the view. Learn how to save Side Panel settings in section 4.6.

2.2 Zoom and selection in View Area

All views except tabular and text views support zooming. Figure 2.13 shows the zoom tools, located at the bottom right corner of the view.

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

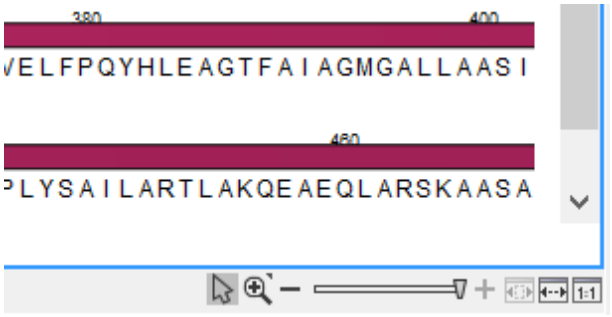


Figure 2.13: *The zoom tools are located at the bottom right corner of the view.*

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

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

2.2.1 Zoom in

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

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

or **Click Zoom in mode ( ) in the zoom tools | click-and-drag a box around a part of the view | the view now zooms in on the part you selected**

or **Press '+' on your keyboard**

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

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

The last option for zooming in is only available if you have a mouse with a scroll wheel:

or **Press and hold Ctrl (⌘ on Mac) | Move the scroll wheel on your mouse forward**

Note! You might have to click in the view before you can use the keyboard or the scroll wheel to zoom.

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

If you want to zoom in to 100 % to see the data at base level, click the **Zoom to base level** ( ) icon.

2.2.2 Zoom out

It is possible to zoom out in different ways:

Click Zoom out mode ( ) in the zoom tools (or press Ctrl+3) | click in the view

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

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

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

The last option for zooming out is only available if you have a mouse with a scroll wheel:

or **Press and hold Ctrl (⌘ on Mac)** | **Move the scroll wheel on your mouse backwards**

Note! You might have to click in the view before you can use the keyboard or the scroll wheel to zoom.

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

If you press **Shift** while clicking in a **View**, the zoom function is reversed. Hence, clicking on a sequence in this way while the **Zoom out** mode toolbar item is selected, zooms in instead of zooming out.

2.2.3 Selecting, panning and zooming

In the zoom tools, you can control which mouse mode to use. The default is **Selection mode** ( ) which is used for selecting data in a view. Next to the selection mode, you can select the **Zoom in mode** as described in section 2.2.1. If you press and hold this button, two other modes become available as shown in figure 2.14:

- **Panning** ( ) is used for dragging the view with the mouse as a way of scrolling.
- **Zoom out** ( ) is used to change the mouse mode so that whenever you click the view, it zooms out.

Figure 2.14: *Additional mouse modes can be found in the zoom tools.*

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

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

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

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

2.3 Toolbox and Status Bar

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

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

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

View | Show/Hide Toolbox

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

2.3.1 Processes

By clicking the **Processes** tab, the Toolbox displays previous and running processes. The running processes can be stopped, paused, and resumed by clicking the small icon ( ) next to the process (see figure 2.15).

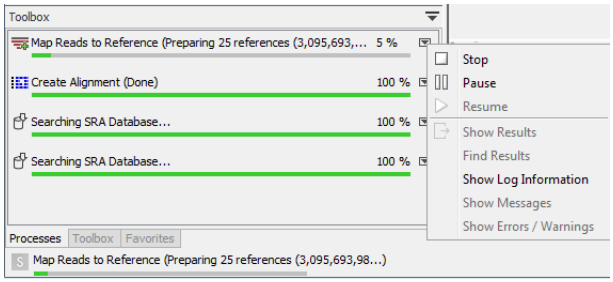


Figure 2.15: *A database search and an alignment calculation are running. Clicking the small icon next to the process allow you to stop, pause and resume processes.*

Stopped and paused processes are not deleted. Processes can be removed by:

View | Remove Finished Processes ( ) .

Besides the options to stop, pause and resume processes, there are some extra options for *a selected number* of the tools running from the Toolbox:

- **Show results**. If you have chosen to save the results (see section 8.2), you will be able to open the results directly from the process by clicking this option.
- **Find results**. If you have chosen to save the results (see section 8.2), you will be able to highlight the results in the Navigation Area.
- **Show Log Information**. This will display a log file showing progress of the process. The log file can also be shown by clicking **Show Log** in the "handle results" dialog where you choose between saving and opening the results.
- **Show Messages**. Some analyses will give you a message when processing your data. The messages are the black dialogs shown in the lower left corner of the Workbench that disappear after a few seconds. You can reiterate the messages that have been shown by clicking this option.

If you close the program while there are running processes, a dialog will ask if you are sure that you want to close the program. Closing the program will stop the process, and it cannot be restarted when you open the program again.

2.3.2 Toolbox

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

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

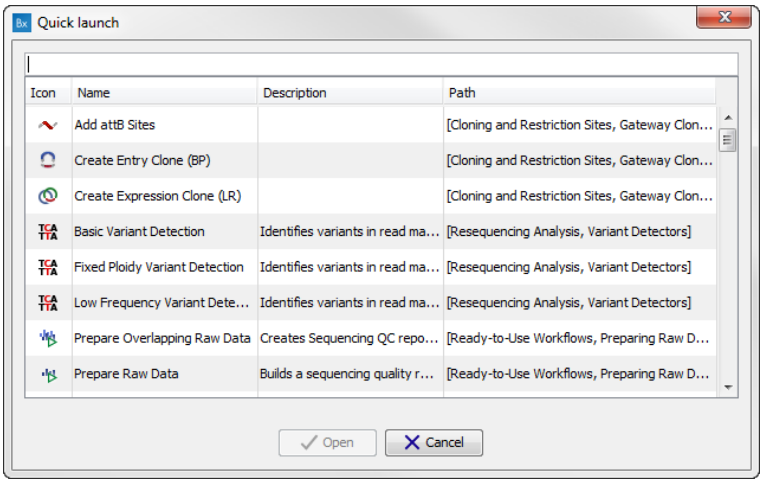


Figure 2.16: *Quick access to all tools in* **Biomedical Genomics Workbench**.

When the dialog is opened, you can start typing search text in the text field at the top. This will bring up the list of tools that match this text either in the name, description or location in the Toolbox. In the example shown in figure 2.17, typing `create` shows a list of tools involving the word "create", and the arrow keys or mouse can be used for selecting and starting a tool.

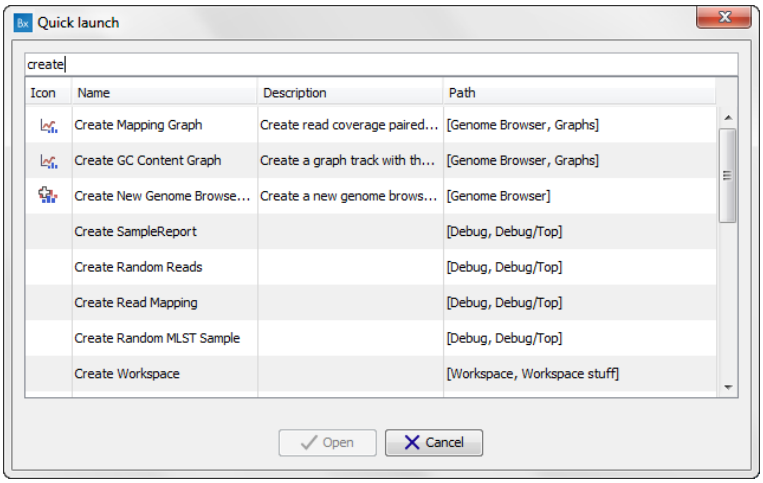


Figure 2.17: *Typing in the search field at the top will filter the list of tools to launch.*

2.3.3 Favorites

Next to the Toolbox tab, you find the **Favorites** tab. This can be used for organizing and getting quick access to the tools you use the most. It consists of two parts as shown in figure 2.18.

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

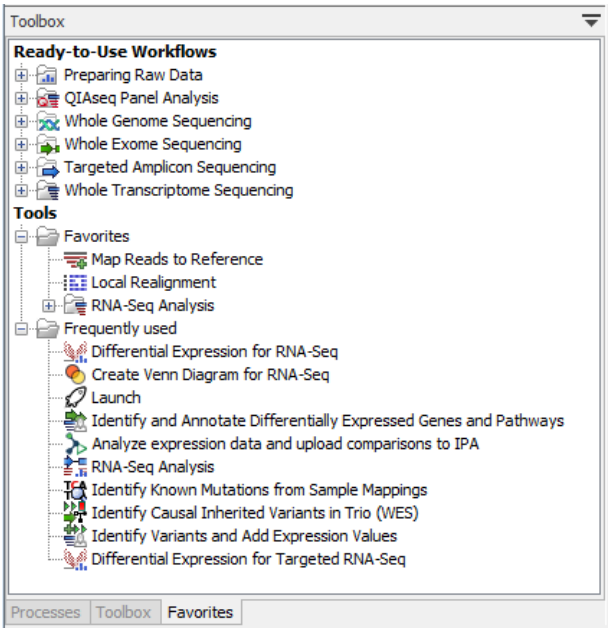


Figure 2.18: *Favorites toolbox.*

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

2.3.4 Status Bar

As can be seen from figure 2.1, the Status Bar is located at the bottom of the window. In the left side of the bar is an indication of whether the computer is making calculations or whether it is idle. The right side of the Status Bar indicates various information depending on the context: it can be the size of a region selected on a sequence, the variant at the position where the cursor stands, or how many rows are selected in a table.

2.4 Workspace

If you are working on a project and have arranged the views for this project, you can save this arrangement using **Workspaces**. A Workspace remembers the way you have arranged the views, and you can switch between different workspaces.

The Navigation Area always contains the same data across workspaces. It is, however, possible to open different folders in the different workspaces. Consequently, the program allows you to display different clusters of the data in separate workspaces.

All workspaces are automatically saved when closing down *Biomedical Genomics Workbench*. The next time you run the program, the workspaces are reopened exactly as you left them.

Note! It is not possible to run more than one version of *Biomedical Genomics Workbench* at a time. Use two or more workspaces instead.

Create Workspace When working with large amounts of data, it might be a good idea to split the work into two or more workspaces. As default the *Biomedical Genomics Workbench* opens

one workspace. Additional workspaces are created in the following way:

Workspace in the Menu Bar | **Create Workspace** | **enter name of Workspace** | **OK**

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

Select Workspace When there is more than one workspace in the *Biomedical Genomics Workbench*, there are two ways to switch between them:

Workspace ( ) in the Toolbar | **Select the Workspace to activate**

or **Workspace in the Menu Bar** | **Select Workspace ( )** | **choose which Workspace to activate** | **OK**

Delete Workspace Deleting a workspace can be done in the following way:

Workspace in the Menu Bar | **Delete Workspace** | **choose which Workspace to delete** | **OK**

Note! Be careful to select the right Workspace when deleting. The delete action cannot be undone. (However, no data is lost, because a workspace is only a representation of data.)

It is not possible to delete the default workspace.

2.5 List of shortcuts

The keyboard shortcuts in *Biomedical Genomics Workbench* are listed below.

Action	Windows/Linux	macOS
Adjust selection	Shift + arrow keys	Shift + arrow keys
Adjust workflow layout	Shift + Alt + L	⌘ + Shift + Alt + L
Back to Navigation Area	Alt + Home	⌘ + Home
	or Alt + fn + left arrow	or ⌘ + fn + left arrow
Close	Ctrl + W	⌘ + W
Close all views	Ctrl + Shift + W	⌘ + Shift + W
Copy	Ctrl + C	⌘ + C
Create track list	Ctrl + L	⌘ + L
Cut	Ctrl + X	⌘ + X
Delete	Delete	Delete or ⌘ + Backspace
Exit	Alt + F4	⌘ + Q
Export	Ctrl + E	⌘ + E
Export graphics	Ctrl + G	⌘ + G
Find Next Conflict	'.' (dot)	'.' (dot)
Find Previous Conflict	',' (comma)	',' (comma)
Help	F1	F1
Import	Ctrl + I	⌘ + I
Launch tools	Ctrl + Shift + T	⌘ + Shift + T
Maximize/restore size of View	Ctrl + M	⌘ + M
Move gaps in alignment	Ctrl + arrow keys	⌘ + arrow keys
New Folder	Ctrl + Shift + N	⌘ + Shift + N
Panning Mode	Ctrl + 4	⌘ + 4
Paste	Ctrl + V	⌘ + V
Print	Ctrl + P	⌘ + P
Redo	Ctrl + Y	⌘ + Y
Rename	F2	F2
Save	Ctrl + S	⌘ + S
Save As	Ctrl + Shift + S	⌘ + Shift + S
Scrolling horizontally	Shift + Scroll wheel	Shift + Scroll wheel
Search local data	Ctrl + Shift + F	⌘ + Shift + F
Search via Side Panel	Ctrl + F	⌘ + F
Select All	Ctrl + A	⌘ + A
Select Selection Mode	Ctrl + 1 (one)	⌘ + 1 (one)
Show folder content	Ctrl + O	⌘ + O
Show/hide Side Panel	Ctrl + U	⌘ + U
Sort folder	Ctrl + Shift + R	⌘ + Shift + R
Split Horizontally	Ctrl + T	⌘ + T
Split Vertically	Ctrl + J	⌘ + J
Switch tabs in View Area	Ctrl + PageUp/PageDown	Ctrl + PageUp/PageDown
	or Ctrl + fn + arrow up/down	or Ctrl + fn + arrow up/down
Switch views	Ctrl + Shift + PageUp/arrow up	Ctrl + Shift + PageUp/arrow up
	Ctrl + Shift + PageDown/arrow down	Ctrl + Shift + PageDown/arrow down
Undo	Ctrl + Z	⌘ + Z
Update folder	F5	F5
User Preferences	Ctrl + K	⌘ + ,

Scroll and Zoom shortcuts

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

Workflows related shortcuts

Action	Windows/Linux	macOS
Workflow, add element	Alt + Shift + E	Alt + Shift + E
Workflow, collapse if its expanded	Alt + Shift + '-' (minus)	Alt + Shift + '-'
Workflow, create installer	Alt + Shift + I	Alt + Shift + I
Workflow, execute	Ctrl + enter	⌘ + enter
Workflow, expand if its collapsed	Alt + Shift + '+' (plus)	Alt + Shift + '-'
Workflow, highlight used elements	Alt + Shift + U	Alt + Shift + U
Workflow, remove all elements	Alt + Shift + R	Alt + Shift + R

Combinations of keys and mouse movements.

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

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

Chapter 3

Data organization

Contents

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

3.1 Navigation Area

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

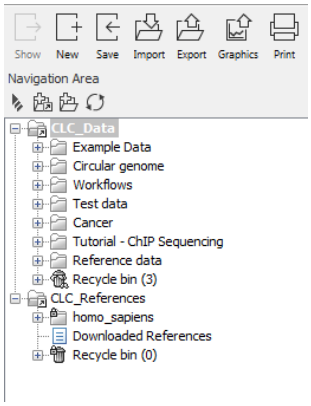


Figure 3.1: *The Navigation Area.*

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

3.1.1 Data structure

The data in the **Navigation Area** is organized into a number of **Locations**. When the *Biomedical Genomics Workbench* is started for the first time, there is one location called *CLC_Data* (unless your computer administrator has configured the installation otherwise).

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

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

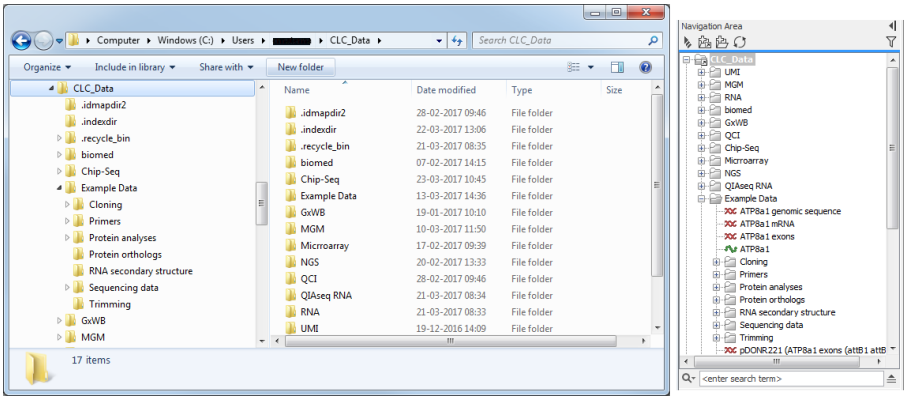


Figure 3.2: *In this example the location called "CLC_Data" points to the folder at C:\Users\<username>\CLC_Data.*

Adding locations

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

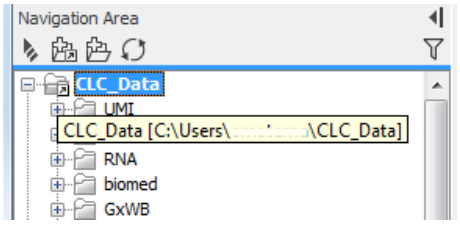


Figure 3.3: *Mousing over the location called 'CLC_Data' shows the full path to the system folder, which in this case is C:\Users\<username>\CLC_Data.*

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

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

File | **New** | **Location ( )**

This will bring up a dialog where you can navigate to the folder you wish to use as your new location (see figure 3.4).

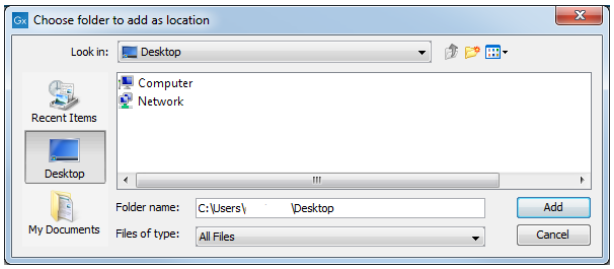


Figure 3.4: *Navigating to a folder to use as a new location.*

When you click **Open**, the new location is added to the **Navigation Area** as shown in figure 3.5.

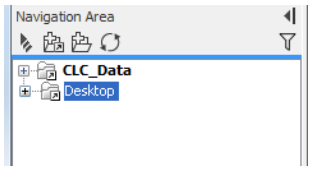


Figure 3.5: *The new location has been added.*

The name of the new location will be the name of the folder selected for the location. To see where the folder is located on your computer, place your mouse cursor on the location icon ( ) for a second. This will show the path to the location.

You can use a folder on a network drive or a removable drive as a Data Location. Such a location will appear inactive if the relevant drive is not available when you start up the Workbench. Once the drive is available, click on Update All (Image updates) and the relevant Data Location will become active (note that there might be a few seconds delay from the moment you connect).

Sharing data is possible when a network drive is available to multiple Workbenches. In this case, you can add the same folder as a Data Location on each Workbench. However, it is important to note that data sharing is not actively supported: we do not support concurrent alteration of data and while the software will often detect this situation and handle it appropriately, by for example

only allowing read access to all but the one party editing the file, we do not guarantee this. In addition, any functionality that involves using the data search indices, (e.g. search functionality, associating metadata with data), will not work properly for shared data locations. Re-indexing a Data Location can help in the short term, but as soon as a new file is created by another piece of software, the index will be out of date. If you decide to share data via Workbenches this way, it is vital that any Workbench that adds a Data Location already used by other Workbenches uses as a Data Location **the exact same folder from the network drive file system hierarchy as the other Workbenches have used**. Indicating a folder higher up or lower down in the hierarchy will cause problems with the indexing of the files, meaning that newly created objects by Workbench A will not be found by Workbench B and vice versa.

Opening data

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

Double-clicking on the element

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

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

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

This will open a view in the **View Area**, which is described in section 2.1.

Adding data

Data can be added to the **Navigation Area** in a number of ways. Files can be imported from the file system (see chapter 6). Furthermore, an element can be added by dragging it into the **Navigation Area**. This could be views that are open, elements on lists, e.g. search hits or sequence lists, and files located on your computer. Finally, you can add data by adding a new location (see section 3.1.1).

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

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

3.1.2 Create new folders

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

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

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

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

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

3.1.3 Sorting folders

You can sort the elements in a folder alphabetically:

right-click the folder | **Sort Folder**

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

3.1.4 Multiselecting elements

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

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

3.1.5 Moving and copying elements

Elements can be moved and copied in several ways:

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

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

Copy, cut and paste functions

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

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

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

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

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

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

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

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

When you have cut the element, it is "grayed out" until you activate the paste function. If you change your mind, you can revert the cut command by copying another element.

Note that if you move data between locations, the original data is kept. This means that you are essentially doing a copy instead of a move operation.

Move using drag and drop

Using drag and drop in the **Navigation Area**, as well as in general, is a four-step process:

click the element | **click on the element again, and hold left mouse button** | **drag the element to the desired location** | **let go of mouse button**

This allows you to:

- Move elements between different folders in the **Navigation Area**
- Drag from the **Navigation Area** to the **View Area**: A new view is opened in an existing **View Area** if the element is dragged from the **Navigation Area** and dropped next to the tab(s) in that **View Area**.
- Drag from the **View Area** to the **Navigation Area**: The element, e.g. a sequence, alignment, search report etc. is saved where it is dropped. If the element already exists, you are asked whether you want to save a copy. You drag from the **View Area** by dragging the tab of the desired element.

Use of drag and drop is supported throughout the program, also to open and re-arrange views (see section 2.1.6).

Note that if you move data between locations, the original data is kept. This means that you are essentially doing a copy instead of a move operation.

Copy using drag and drop

To copy instead of move using drag and drop, hold the Ctrl (⌘ on Mac) key while dragging:

click the element | **click on the element again, and hold left mouse button** | **drag the element to the desired location** | **press Ctrl (⌘ on Mac) while you let go of mouse button release the Ctrl/⌘ button**

3.1.6 Change element names

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

Change how sequences are displayed

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

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

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

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

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

Rename element

Renaming a folder or an element in the **Navigation Area** can be done in two different ways:

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

or **select the element** | **F2**

When you can rename the element, you can see that the text is selected and you can move the cursor back and forth in the text. When the editing of the name has finished; press **Enter** or select another element in the **Navigation Area**. If you want to discard the changes instead, press the **Esc**-key.

3.1.7 Delete, restore and remove elements

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

Deleting a folder or an element from a Workbench data location can be done in two ways:

right-click the element | Delete ( )

or **select the element | press Delete key**

This will cause the element to be moved to the **Recycle Bin** ( ) where it is kept until the recycle bin is emptied or until you choose to restore the data object to your data location.

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

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

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

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

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

Edit in the Menu Bar | Empty Recycle Bin ( )

This deletes the data and frees up disk space.

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

3.1.8 Show folder elements in a table

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

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

or

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

An example is shown in figure 3.6.

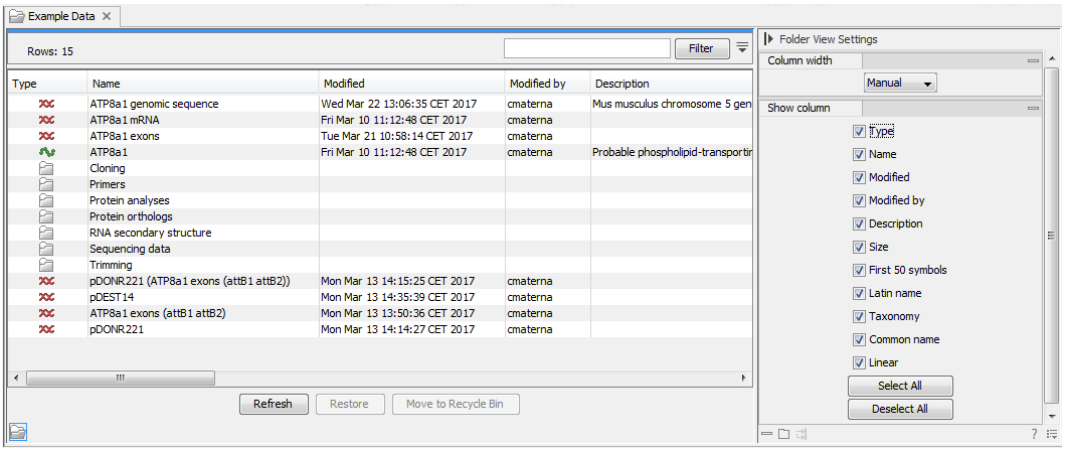


Figure 3.6: *Viewing the elements in a folder.*

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

heading of another column.

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

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

Batch edit folder elements

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

In figure 3.7 you can see an example where the name of two sequence are renamed in one go. In this example, a dialog with a text field will be shown, letting you enter a new name for these two sequences.

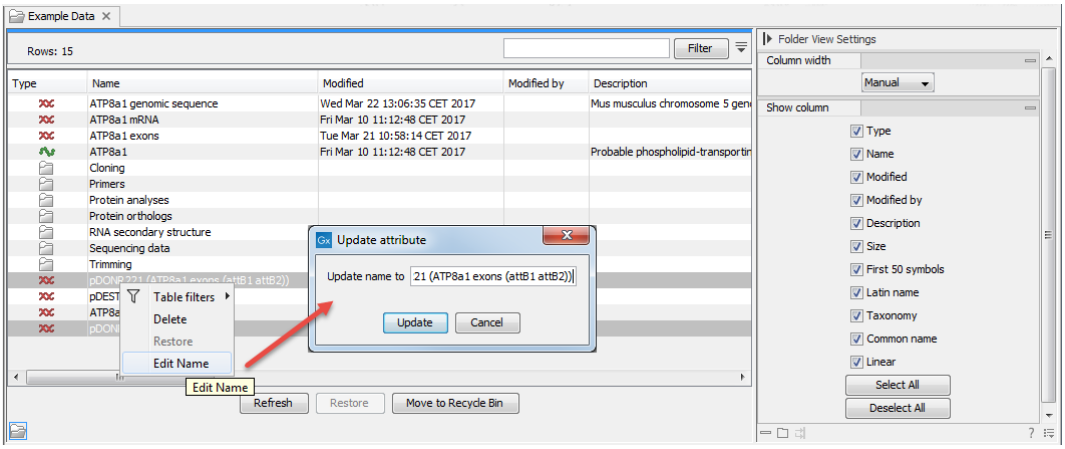


Figure 3.7: *Changing the common name of two sequences.*

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

Drag and drop folder elements

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

3.2 Metadata

Metadata refers to information about data. In the context of the CLC Workbenches, this will usually mean information about samples. For example a set of reads could come from a particular specimen at a particular time point with particular characteristics. The specimen, time and characteristics would be metadata for that set of reads. The data can then be associated

with its metadata in the Workbench. This can be useful for keeping track of related datasets and metadata can be used by some types of analyses in some CLC Workbenches.

Metadata can be created directly in the Workbench, but typically it will be imported from an external file (excel or text based). See section 3.2.1. It is then stored as a metadata table in the Workbench. An example of a metadata table as it might appear in the Workbench is shown in figure 3.8.

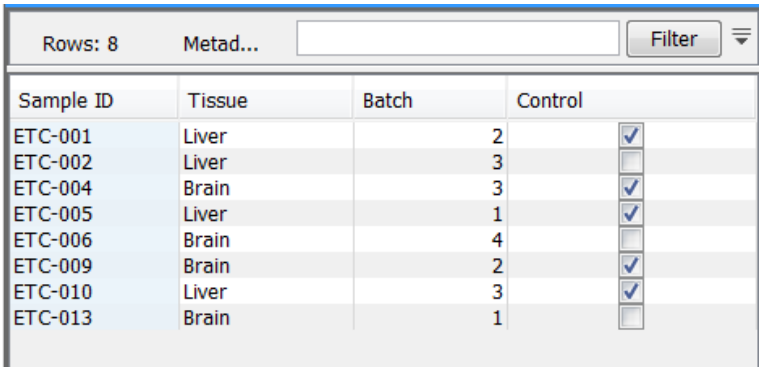


Figure 3.8: *A simple Metadata Table.*

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

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

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

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

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

3.2.1 Importing Metadata

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

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

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

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

To run the basic importer, go to:

File | Import () | **Import Metadata** ()

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

The rows in the spreadsheet are displayed in the Metadata preview window, as shown in figure 3.9. Click **Next**.

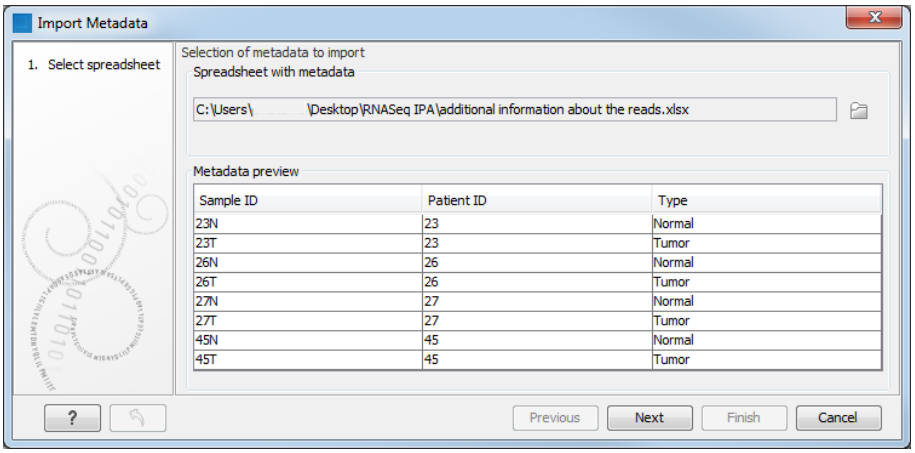


Figure 3.9: *After an Excel file is selected, its rows are visible in the Metadata preview table.*

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

Associating data with the metadata being imported is illustrated in figure 3.10. To do this:

- In the field labeled **Location of data**, click on the folder icon to the right and select the data elements of interest.

- In the Matching scheme section, select whether data element names must match exactly the entries in the first column of the metadata to have an association created (Exact), or whether partial matches are allowed (Partial). The two matching schemes are described in detail in section 3.2.3.

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

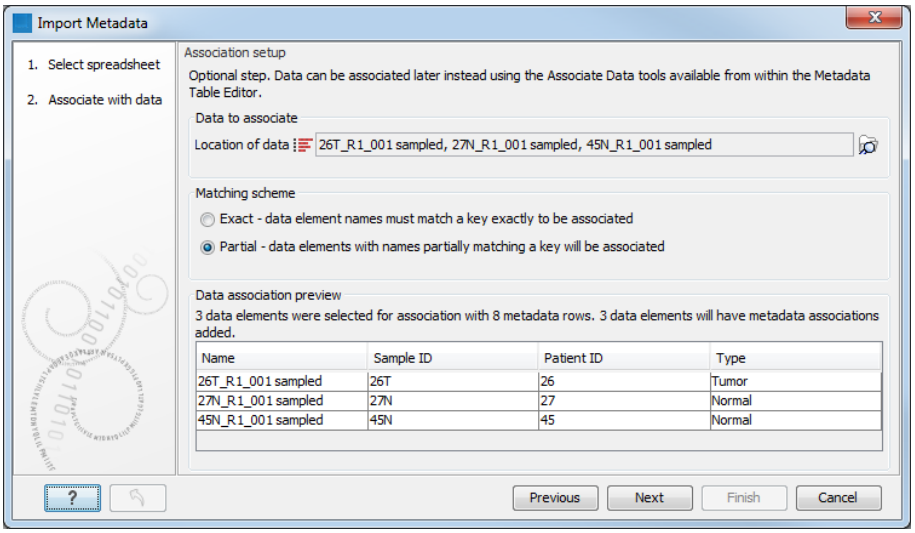


Figure 3.10: *Three data elements were selected for association with 8 metadata rows. All three will have an associated added. Here, the partial matching scheme has been selected.*

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

The associated information can be viewed for a given data element in the Show Element Info view, as show in figure 3.11.

3.2.2 Advanced Metadata Import

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

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

- Can import from Excel (.xlsx/.xls) or text files with a common delimiter can be used.
- The structure of the metadata table (the columns, their type, and the key column) must be set up before the metadata (contents) are imported.

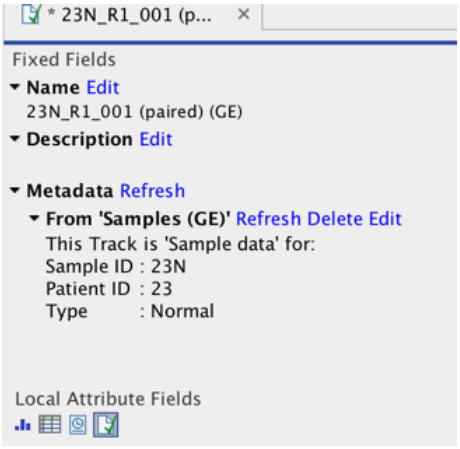


Figure 3.11: *Metadata associations can be seen, edited, refreshed or deleted via the Show Element Info view.*

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

To start the Metadata Table Editor, go to:

File | **New** | **Metadata Table (** **)**

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

Defining the table structure

Click on the button labeled **Setup Table** at the bottom of the view (figure 3.12).

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 3.13), 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.

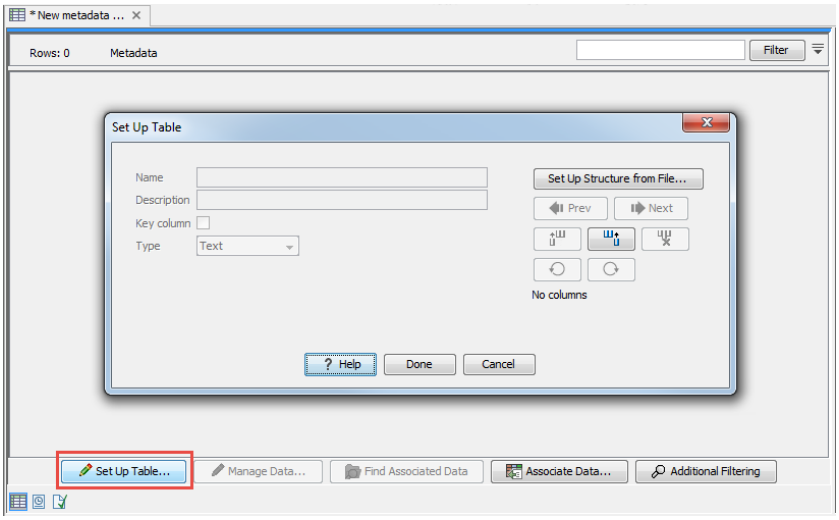


Figure 3.12: *Dialog used to add columns to an empty Metadata Table.*

- **Encoding** For text files only: the encoding used to create the file. The default is UTF-8.
- **Separator** For text files only: The character used to separate the columns. The default is semicolon (;).

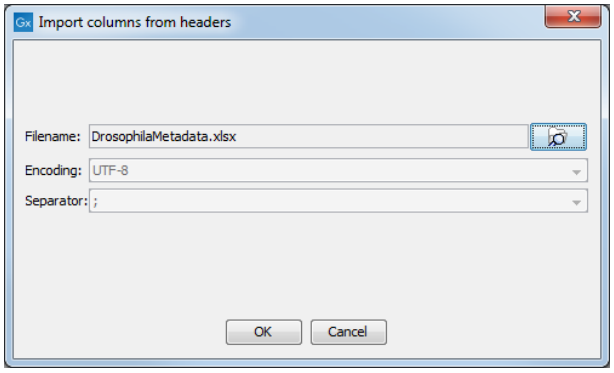


Figure 3.13: *Creating a metadata table structure based on an external file.*

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

Edit the following information for each column:

- **Name**. A mandatory header name or title for the column.
- **Description**. An optional description of the information that will be held in the column. The description will appear as a tool tip, visible when you hover the mouse cursor over the column name in the metadata table.
- **Key column**. Put a check in the box in the one column that will be the "key" column. All rows in this column must be populated and all entries in this column must be unique.
- **Type**. The type of value allowed. The available types are:
  - **Text** Simple text.

- **Whole number** Integer values, like `42` or `-7`.
- **Decimal number** Decimal values, like `3.14` or `1.72e13`.
- **Yes / No** `Yes/No` or `True/False` values are accepted. Capitalization is not necessary.
- **Date** Local dates such as `2015-04-23` for April 23rd, 2015.
- **Date and time** Local date and time such as `2015-04-23 13:37` for 1:37pm on April 23rd, 2015. Note the use of 24-hour clock and that no time zone information is present.

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

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

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

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

Populating the table

Click on **Manage Data** button at the bottom of the view (figure 3.14).

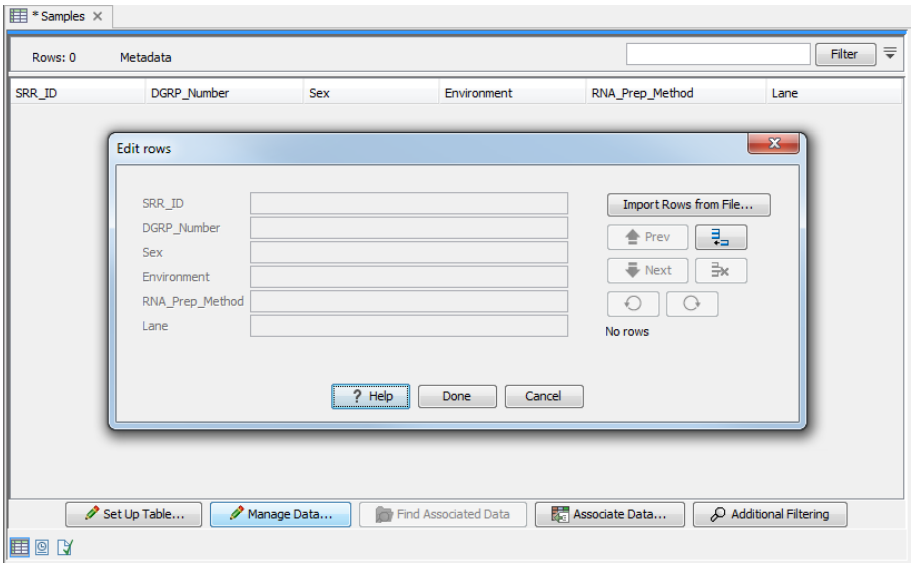


Figure 3.14: *Tool for managing the metadata itself. Notice the button labeled Import Rows from File.*

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

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

Click on **Import Rows from File** and select the external file of metadata. This brings up the window shown in figure 3.15.

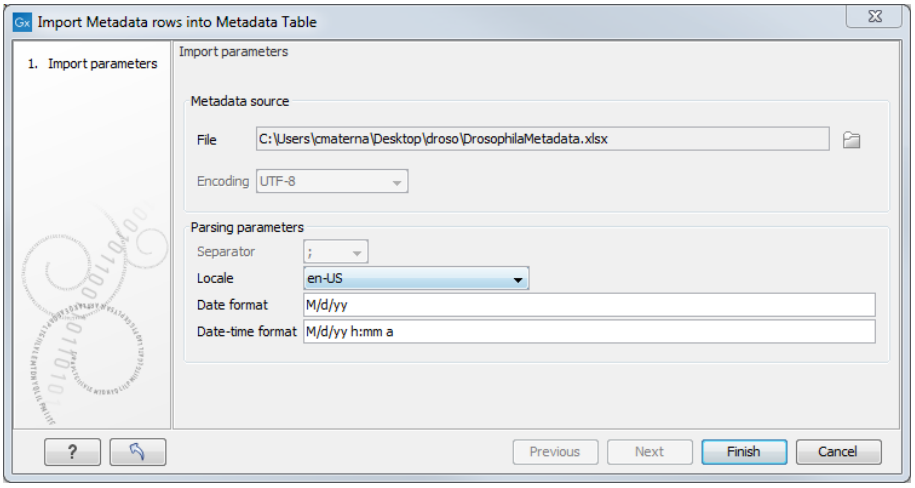


Figure 3.15: *Tool to import rows into a Metadata Table.*

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 seledcted file. Specifying the correct encoding is important to ensure that the file is correctly interpreted.
- **Separator**. For text files only: the character used to separate columns in the file.
- **Locale**. For text files only: the locale used to format numbers and dates within the file.
- **Date format**. For text files only: the date format used in the imported file.
- **Date-time format**. For text files only: the date-time format used in the imported file.
  The date and date-time templates uses the Java patterns for date and time formatting.
  Meaning of some of the symbols:

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

Examples of using this:

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

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

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

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

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

3.2.3 Associating data elements with metadata

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

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

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

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

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

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

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

Associate Data Automatically

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

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

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

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

Select the data the tool should consider when setting up metadata associations in the window that appears. An example of this is shown in figure 3.16. You can select an item or sets of items in the navigation area on the left and move these into the selected elements list. Alternatively, you can right click on a folder and specify that all elements in the folder should be put in the selected elements list. This is illustrated in figure 3.17.

.

Specify a role that should be assigned to each data element that is associated to a metadata row (figure 3.18). The role can be anything that describes the data element best.

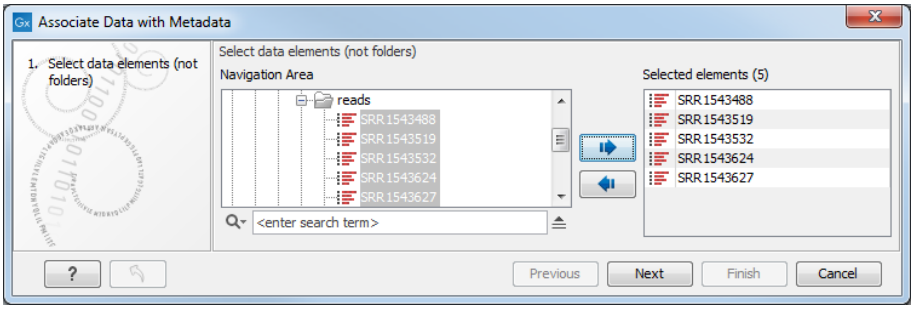


Figure 3.16: *Select data for automatic metadata association.*

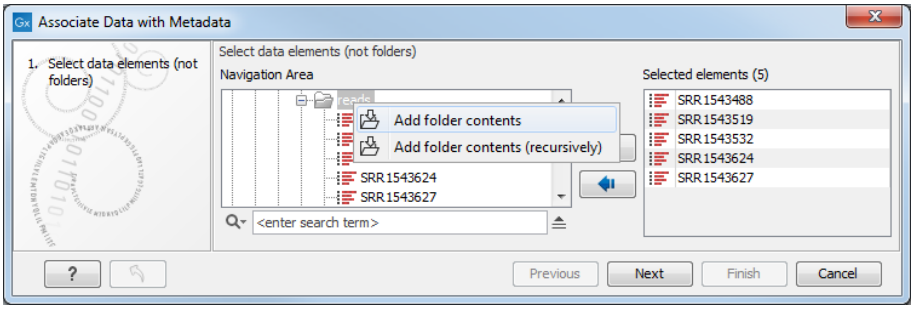


Figure 3.17: *Selecting all data elements in a folder.*

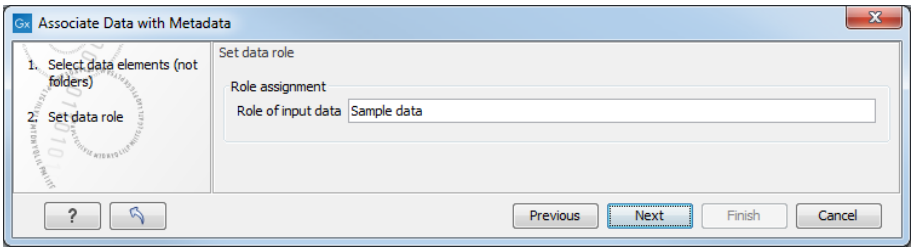


Figure 3.18: *Provide a role for the data elements. The default role provided is "Sample data".*

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

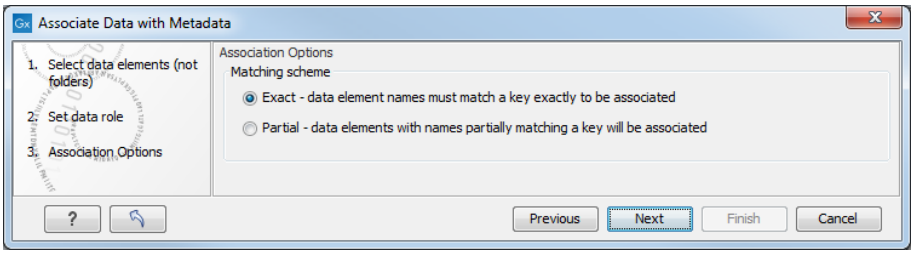


Figure 3.19: *Data element names can be matched either exactly or partially to the entries in the key column.*

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

Warning: It is safest only to select data elements that have no existing association to the metadata table being worked with, or carefully selecting any data elements with an existing association which you wish to update. All data selected that has an association with the metadata table being worked with will be *updated* by the automatic association tool. This means that any new or updated information in a metadata row can be added, but it also means that if

no rows in the metadata match such a data element anymore, then the data association will be removed. This could happen if, for example, you changed the name of a data element with a metadata association, and did not change the corresponding key entry in the metadata table.

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

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

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

The following describes the matching process in detail:

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

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

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

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

  For example, a key entry `Sample234` would be a match to the data element with name `Sample234-1 (mapped) (trimmed)` because the whole key entry matches the whole of the first component of the data element name. Conversely, if 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 metadat row the data will be associated with.

  The table below provides examples to illustrate the partial matching system, on a table that has the keys with sample IDs like in figure 3.20) (i.e. `ETC-001`, `ETC-002`, ..., `ETC-013`),

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

Associate Data with Row

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

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

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

- Select the desired row in the metadata table by clicking on it.
- Right click and select the **Associate Data with Row** option (see figure 3.20), or click on the **Associate Data** button at the bottom of the view and choose the option **Associate Data with Row**.

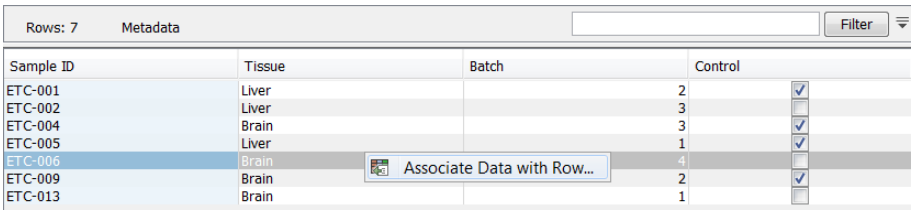


Figure 3.20: *Manual association of data elements to a metadata row.*

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

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

- Click on the button labeled **Next**.
- Enter a role for the data elements that have been chosen.
- Click on the button labeled **Next**.
- Click on the button labeled **Next** and then choose to **Save** the outputs.

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

3.2.4 Working with data and metadata

Finding data elements based on metadata

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

To find data elements associated with selected metadata rows:

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

  A table with a listing of the data elements associated to the selected metadata row(s) will appear (figure 3.21).

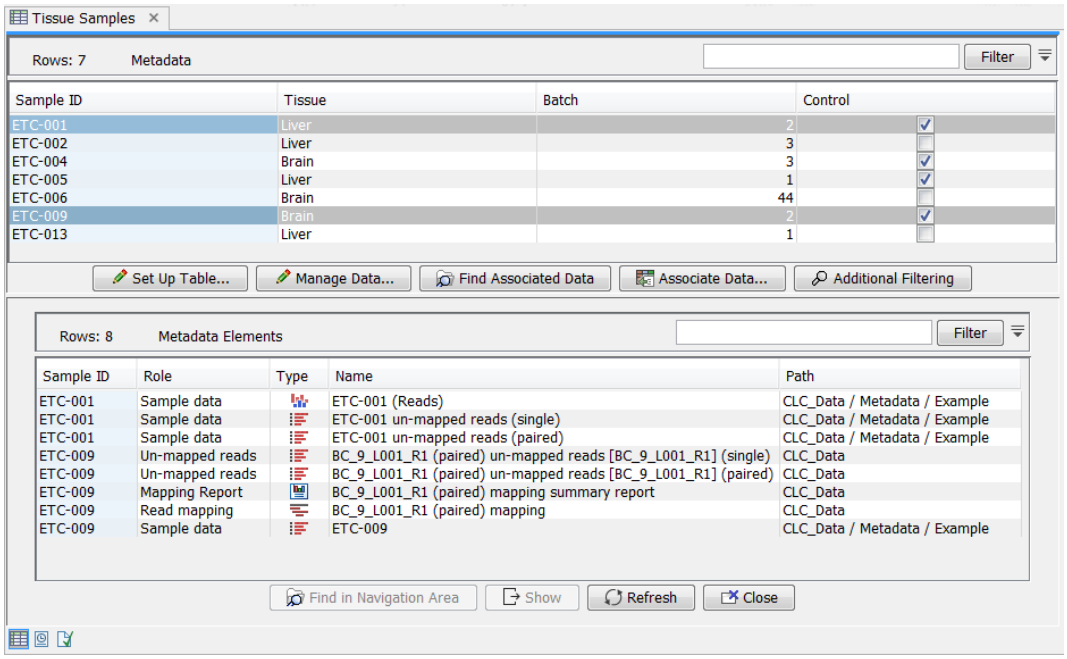


Figure 3.21: *Metadata Table with search results*

The search results table shows the type, name, and navigation area path for each data element found. It also shows the key entry of the metadata table row with which the element is associated and the role of the data element for this metadata association. In figure 3.21, there are five data elements associated with sample ETC-009. Three are Sequence Lists, two of which have a role that tells us that they are un-mapped reads resulting from the Map Reads to Reference tool.

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

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

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

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

Analyses can be launched on the selected data elements:

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

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

Identifying metadata rows without associated data

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

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

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

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

Viewing metadata associations

Metadata associations for a data element are shown in the Element Info view (section 10.4), see figure 3.22. To show Element Info,

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

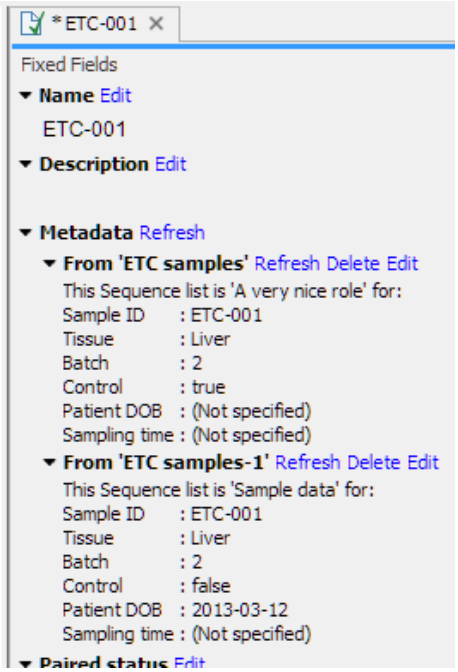


Figure 3.22: *Element Info view with a metadata association*

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.

Removing metadata associations

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

1. Open the metadata table containing the rows of interest.
2. Highlight the relevant rows of the metadata table.
3. Click on the button labeled Find Associated Data.
4. In the Metadata Elements table that opens, highlight the rows for the data elements the metadata associations should be removed from.
5. Right click over the highlighted area and choose the option Remove Association(s) (figure 3.23). Alternatively, use the Delete key on the keyboard, or on a Mac, the fn and backspace keys at the same time.

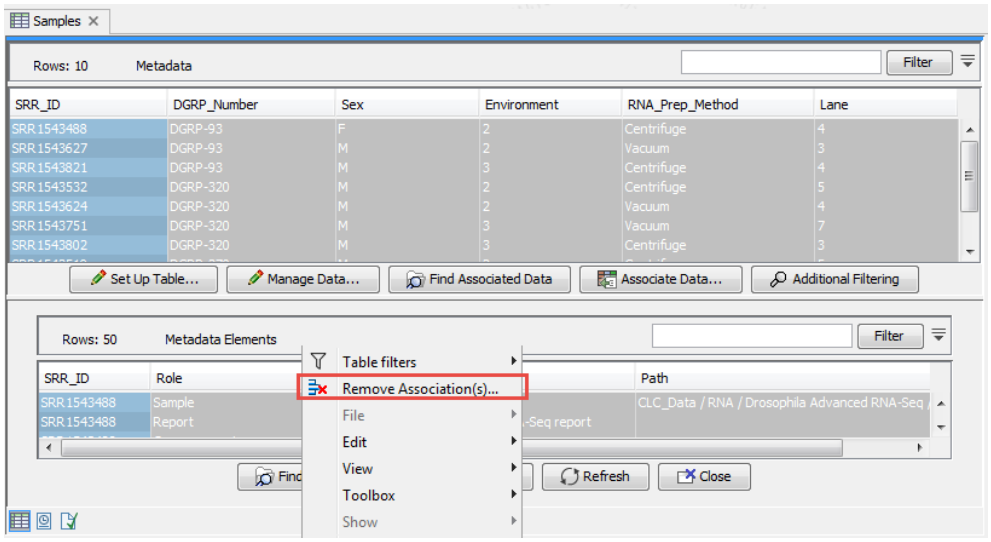


Figure 3.23: *Removing metadata associations to two data elements via the Metadata Elements table.*

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

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

Exporting metadata

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

See section 6.6 for more information.

3.3 Working with tables

Tables are used in a lot of places in the *Biomedical Genomics Workbench*. There are some general features for all tables, irrespective of their contents, that are described here.

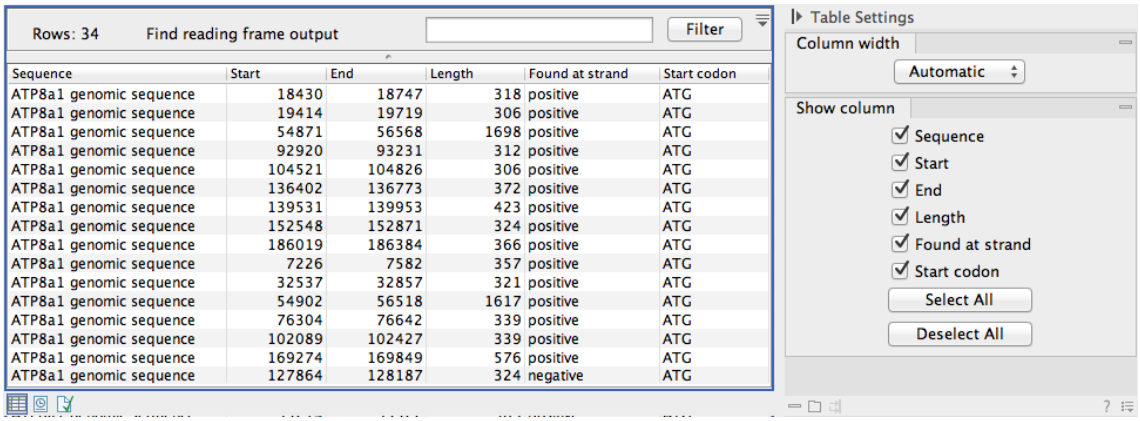
Figure 3.24 shows an example of a typical table. This is the table result of **Find Open Reading Frames** ( ). We use this table as an example to illustrate concepts relevant to all kinds of tables.

Table viewing options

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

For example, the columns that can be dispalyed in the table are listed in the section called **Show column**. The checkboxes allow you to see or hide any of the available columns for that table.

The Column width can be set to **Automatic** or **Manual**. By default, the first time you open a table, it will be set to **Automatic**. The default selected columns are hereby resized to fit the width of the viewing area. When changing to the **Manual** option, column widths will adjust to the actual header size, and each column size can subsequently by adjusted manually. When the table



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

Figure 3.24: *A table showing the results of an open reading frames analysis.*

content exceeds the size of the viewing area, a horizontal scroll becomes available for navigation across the columns.

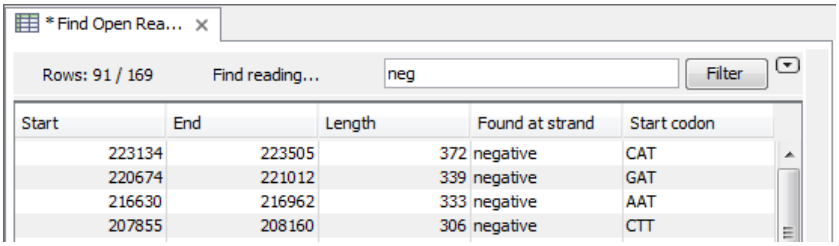
Sorting tables

You can **sort** table according to the values of a particular column by clicking a column header. (Pressing Ctrl - ⌘ on Mac - while you click will refine the existing sorting).

Clicking once will sort in ascending order. A second click will change the order to descending. A third click will set the order back its original order.

3.3.1 Filtering tables

The final concept to introduce is **Filtering**. The table filter as an advanced and a simple mode. The simple mode is the default and is applied simply by typing text or numbers (see an example in figure 3.25).[1]



Start	End	Length	Found at strand	Start codon
223134	223505	372	negative	CAT
220674	221012	339	negative	GAT
216630	216962	333	negative	AAT
207855	208160	306	negative	CTT

Figure 3.25: *Typing "neg" in the filter in simple mode.*

Typing "neg" in the filter will only show the rows where "neg" is part of the text in any of the columns (also the ones that are not shown). The text does not have to be in the beginning, thus "ega" would give the same result. This simple filter works fine for fast, textual and non-complicated filtering and searching.

However, if you wish to make use of numerical information or make more complex filters, you can switch to the advanced mode by clicking the **Advanced filter** ( ) button. The advanced filter is structure in a different way: First of all, you can have more than one criterion in the filter. Criteria can be added or removed by clicking the **Add** ( ) or **Remove** ( ) buttons. At the top, you can

[1] Note that for tables with more than 10000 rows, you have to actually click the **Filter** button for the table to take effect.

choose whether all the criteria should be fulfilled (**Match all**), or if just one of the needs to be fulfilled (**Match any**).

For each filter criterion, you first have to select which **column** it should apply to.

Next, you choose an **operator**. For numbers, you can choose between:

- **=** (equal to)
- **<** (smaller than)
- **>** (greater than)
- **<>** (not equal to)
- **abs. value <** (absolute value smaller than. This is useful if it doesn't matter whether the number is negative or positive)
- **abs. value >** (absolute value greater than. This is useful if it doesn't matter whether the number is negative or positive)

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

For text-based columns, you can choose between:

- **starts with** (the text starts with your search term)
- **contains** (the text does not have to be in the beginning)
- **doesn't contain**
- **=** (the whole text in the table cell has to match, also lower/upper case)
- $\neq$ (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.

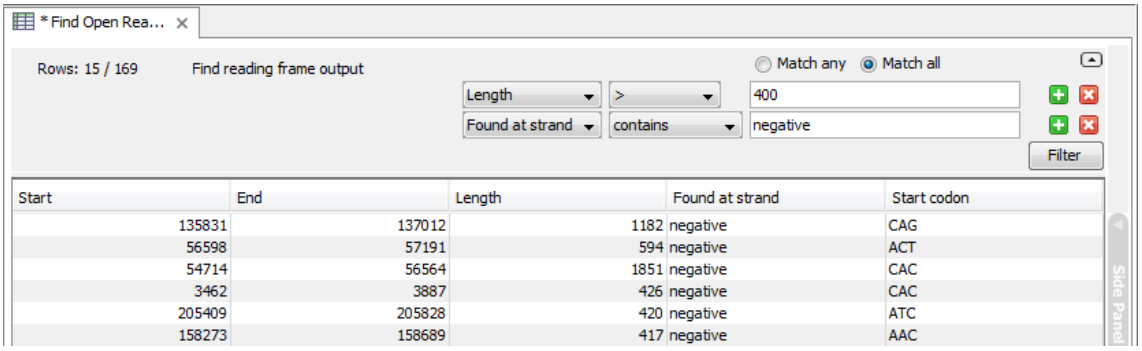


Figure 3.26: *The advanced filter showing open reading frames larger than 400 that are placed on the negative strand.*

Figure 3.26 shows an example of an advanced filter which displays the open reading frames larger than 400 that are placed on the negative strand.

Both for the simple and the advanced filter, there is a counter at the upper left corner which tells you the number of rows that pass the filter (91 in figure 3.25 and 15 in figure 3.26).

3.4 Customized attributes on data locations

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

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

To configure which fields that should be available[2] go to the Workbench:

right-click the data location | Location | Attribute Manager

This will display the dialog shown in figure 3.27.

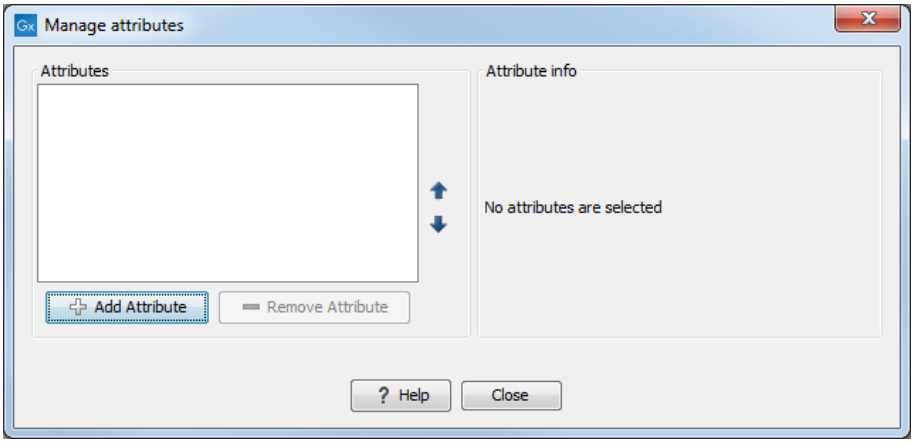


Figure 3.27: *Adding attributes.*

Click the **Add Attribute** ( ) button to create a new attribute. This will display the dialog shown in figure 3.28.

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

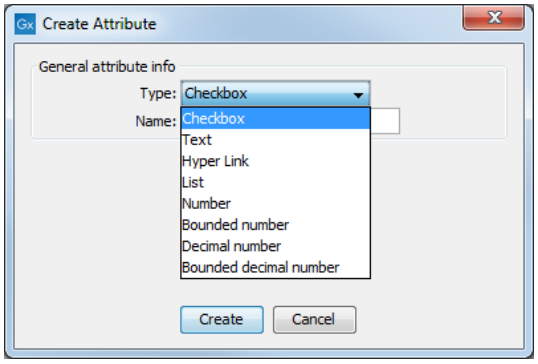


Figure 3.28: *The list of attribute types.*

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 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 3.29).

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

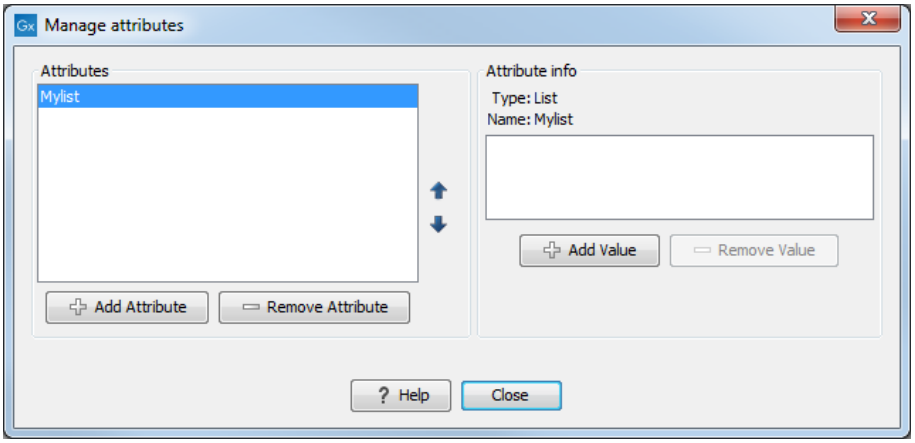


Figure 3.29: *Defining items in a list.*

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

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

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

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

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

3.4.1 Filling in values

When a set of attributes has been created (as shown in figure 3.30), 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 3.31.

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.

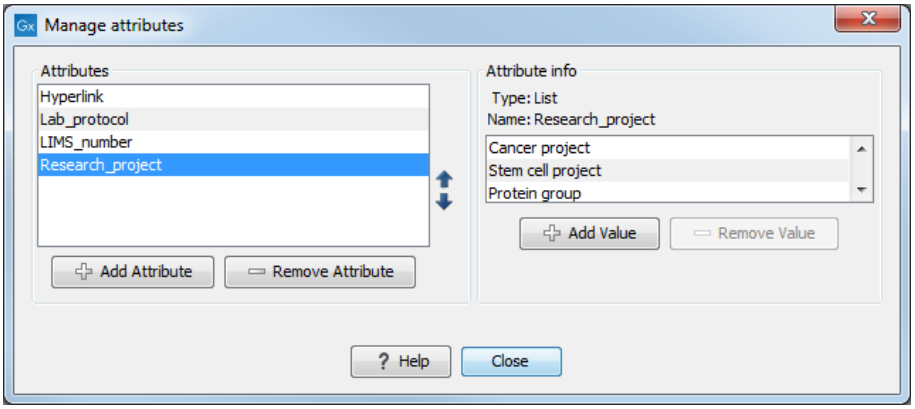


Figure 3.30: *A set of attributes defined in the attribute manager.*

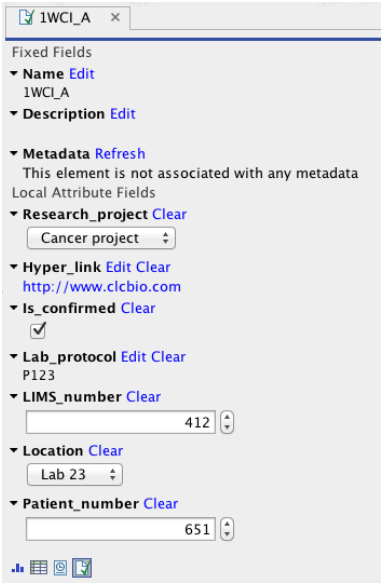


Figure 3.31: *Adding values to the attributes.*

When nobody has entered information, the attribute will have a "Not set" written in red next to the attribute (see figure 3.32).

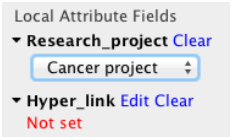


Figure 3.32: *An attribute which has not been set.*

This is particularly useful for attribute types like checkboxes and lists where you cannot tell, from the displayed value, if it has been set or not. Note that when an attribute has not been set, you cannot search for it, even if it looks like it has a value. In figure 3.32, you will *not* be able to find this sequence if you search for research projects with the value "Cancer project", because it has not been set. To set it, simply click in the list and you will see the red "Not set" disappear.

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

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

3.4.2 What happens when a clc object is copied to another data location?

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

If the element (e.g. sequence) is moved back to the original data location, the information will again be editable and searchable.

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

3.4.3 Searching

When an attribute has been created, it will automatically be available for searching. This means that in the **Local Search** ( ), you can select the attribute in the list of search criteria (see figure 3.33).

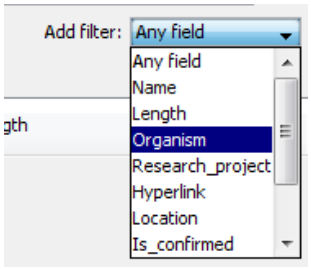


Figure 3.33: *The attributes from figure 3.30 are now listed in the search filter.*

It will also be available in the **Quick Search** below the **Navigation Area** (press Shift+F1 (Fn+Shift+F1 on Mac) and it will be listed - see figure 3.34).

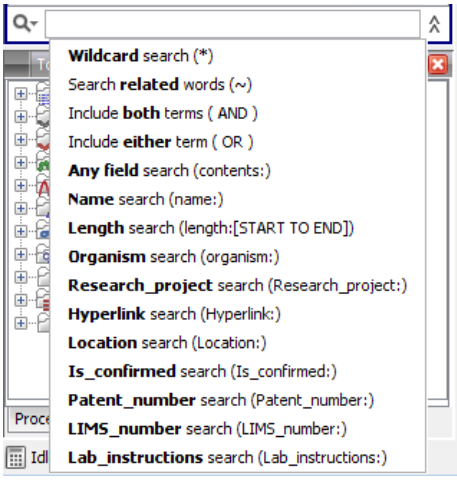


Figure 3.34: *The attributes from figure 3.30 are now available in the Quick Search as well.*

3.5 Local search

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

- **Quick-search** directly from the search field in the **Navigation Area**.
- **Advanced search** which makes it easy to make more specific searches.

In most cases, quick-search will find what you need, but if you need to be more specific in your search criteria, the advanced search is preferable.

What kind of information can be searched? Below is a list of the different kinds of information that you can search for (applies to both quick-search and the advanced search).

- **Name.** The name of a sequence, an alignment or any other kind of element. The name is what is displayed in the **Navigation Area** per default.
- **Length.** The length of the sequence.
- **Organism.** Sequences which contain information about organism can be searched. In this way, you could search for e.g. *Homo sapiens* sequences.
- **Custom attributes.** Read more in section 3.4

Only the first item in the list, **Name**, is available for all kinds of data. The rest is only relevant for sequences.

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

However, if you experience problems with your search results, i.e., if you do not get the hits you expect, it might be because of an index error.

The *Biomedical Genomics Workbench* automatically maintains an index of all data in all locations in the **Navigation Area**. If this index becomes out of sync with the data, you will experience problems with strange results. In this case, you can rebuild the index:

Right-click the relevant location | **Location** | **Rebuild Index**

This will take a while depending on the size of your data. At any time, the process can be stopped in the process area, see section 2.3.1.

3.5.1 Quick search

At the bottom of the **Navigation Area** there is a text field as shown in figure 3.35). To search, simply enter a text to search for and press **Enter**.

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

To show the results, the search pane is expanded.

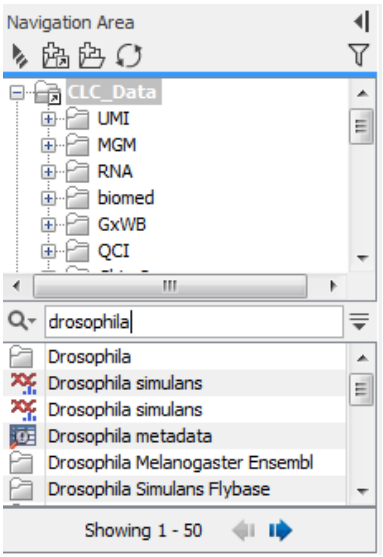


Figure 3.35: *Search simply by typing in the text field and press Enter.*

If there are many hits, only the 50 first hits are immediately shown. At the bottom of the pane you can click **Next** ( ) to see the next 50 hits. In the preferences (see Chapter 4), you can specify the number of hits to be shown.

If a search gives no hits, you will be asked if you wish to search for matches that start with your search term. If you accept this, an asterisk (*) will be appended to the search term.

Pressing the Alt key while you click a search result will highlight the search hit in its folder in the **Navigation Area**.

Search for data locations The search function can also be used to search for a specific URL. This can be useful if you work on a server and wish to share a data location with another user. A simple example is shown in figure 3.36. Right click on the object name in the **Navigation Area** (in this case ATP8a1 genomic sequence) and select "Copy". When you use the paste function in a destination outside the Workbench (e.g. in a text editor or in an email), the data location will become visible. The URL can now be used in the search field in the Workbench to locate the object.

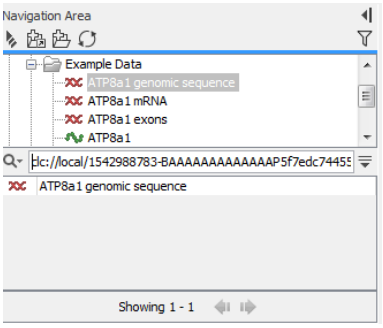


Figure 3.36: *The search field can also be used to search for data locations.*

Quick search history You can access the 10 most recent searches by clicking the icon ( ) next to the search field (see figure 3.37).

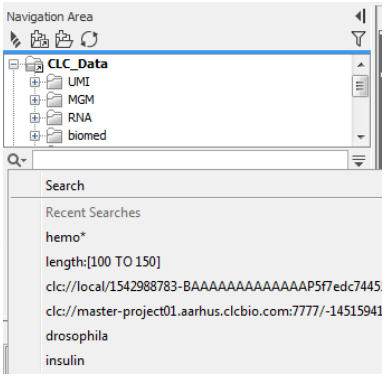


Figure 3.37: *Recent searches.*

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

Special search expressions

When you write a search term in the search field, you can get help to write a more advanced search expression by pressing **Shift+F1**. This will reveal a list of guides as shown in figure 3.38.

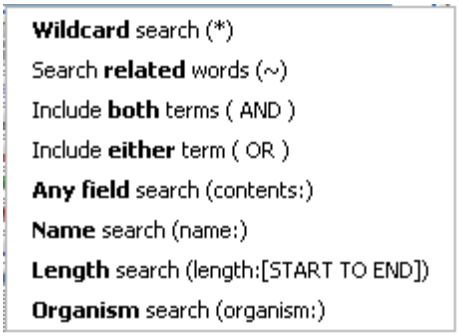


Figure 3.38: *Guides to help create advanced search expressions.*

You can select any of the guides (using mouse or keyboard arrows), and start typing. If you e.g. wish to search for sequences named BRCA1, select "Name search (name:)", and type "BRCA1". Your search expression will now look like this: "name:BRCA1".

The guides available are these:

- **Wildcard search (*)**. Appending an asterisk * to the search term will find matches starting with the term. E.g. searching for "brca*" will find both *brca1* and *brca2*.
- **Search related words (~)**. If you don't know the exact spelling of a word, you can append a tilde to the search term. E.g. "brac1~" will find sequences with a *brca1* gene.
- **Include both terms (AND)**. If you write two search terms, you can define if your results have to match both search terms by combining them with AND. E.g. search for "brca1 AND human" will find sequences where *both* terms are present.
- **Include either term (OR)**. If you write two search terms, you can define that your results have to match either of the search terms by combining them with OR. E.g. search for "brca1 OR brca2" will find sequences where *either* of the terms is present.

- **Do not include term (NOT)** If you write a term after not, then elements with these terms will not be returned.
- **Name search (name:)**. Search only the name of element.
- **Organism search (organism:)**. For sequences, you can specify the organism to search for. This will look in the "Latin name" field which is seen in the **Sequence Info** view (see section 10.4).
- **Length search (length:[START TO END])**. Search for sequences of a specific length. E.g. search for sequences between 1000 and 2000 residues: "length:1000 TO 2000".

Note! If you have added attributes (see section 3.4), these will also appear on the list when pressing **Shift+F1**.

If you do not use this special syntax, you will automatically search for both name, description, organism, etc., and search terms will be combined as if you had put OR between them.

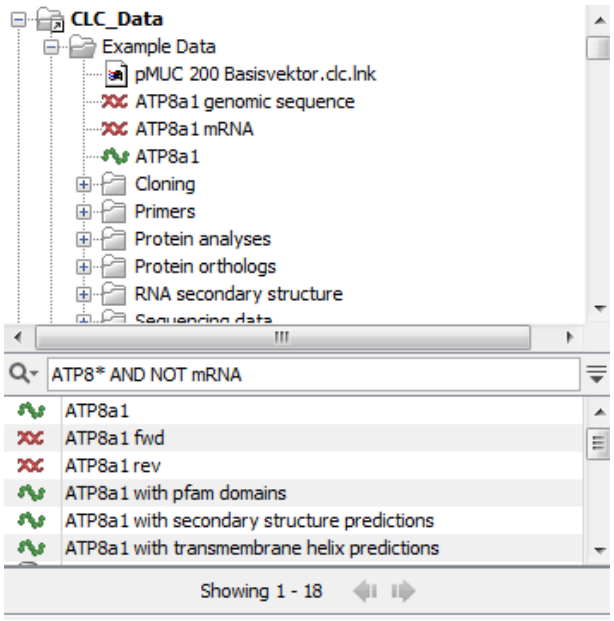


Figure 3.39: *An example of searching for elements with the name, description and organsim information that includes "ATP8" but do not include the term "mRNA".*

3.5.2 Advanced search

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

Edit | Local Search ( )

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

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

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

- All sequences
- Nucleotide sequences
- Protein sequences
- All data

When searching for sequences, you will also get alignments, sequence lists etc as result, if they contain a sequence which match the search criteria.

Below are the search criteria. First, select a relevant search filter in the **Add filter:** list. For sequences you can search for

- Name
- Length
- Organism

See section 3.5.1 for more information on individual search terms.

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

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

- Description
- Keywords
- Common name
- Taxonomy name

To see this information for a sequence, switch to the **Element Info** ( ) view (see section 10.4).

For each search line, you can choose if you want the exact term by selecting "is equal to" or if you only enter the start of the term you wish to find (select "begins with").

An example is shown in figure 3.40.

This example will find human nucleotide sequences (organism is *Homo sapiens*), and it will only find sequences shorter than 10,000 nucleotides.

Note that a search can be saved ( ) for later use. You do not save the search results - only the search parameters. This means that you can easily conduct the same search later on when your data has changed.

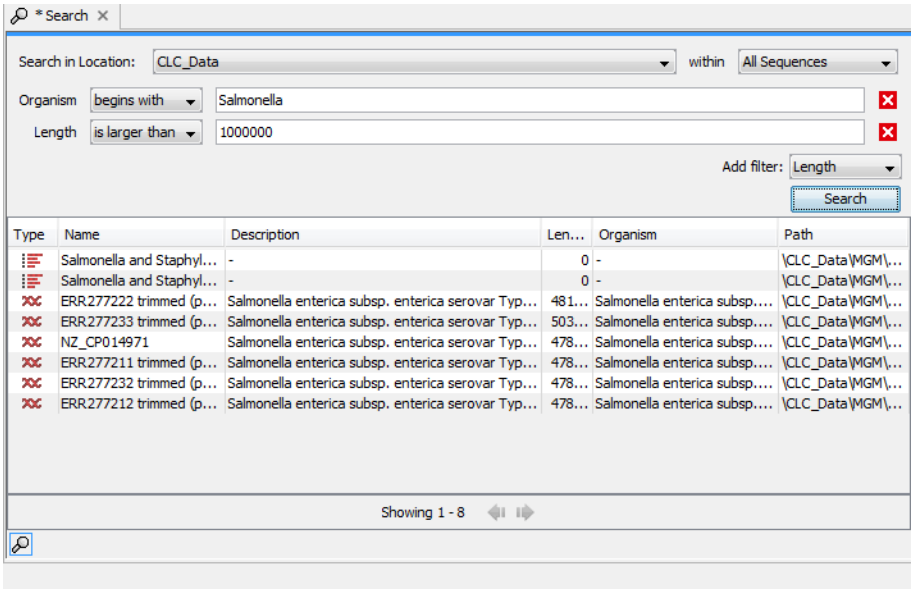


Figure 3.40: *Searching for Salmonella sequences larger than 1 million nucleotides.*

Chapter 4

User preferences and settings

Contents

The first three sections in this chapter deal with the general preferences that can be set for *Biomedical Genomics Workbench* using the **Preferences** dialog. The next section explains how the settings in the **Side Panel** can be saved and applied to other views. Finally, you can learn how to import and export the preferences.

The **Preferences** dialog offers opportunities for changing the default settings for different features of the program.

The **Preferences** dialog is opened in one of the following ways and can be seen in figure 4.1:

Edit | Preferences (⚙)

or **Ctrl + K (⌘ + ; on Mac)**

4.1 General preferences

The **General preferences** include:

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

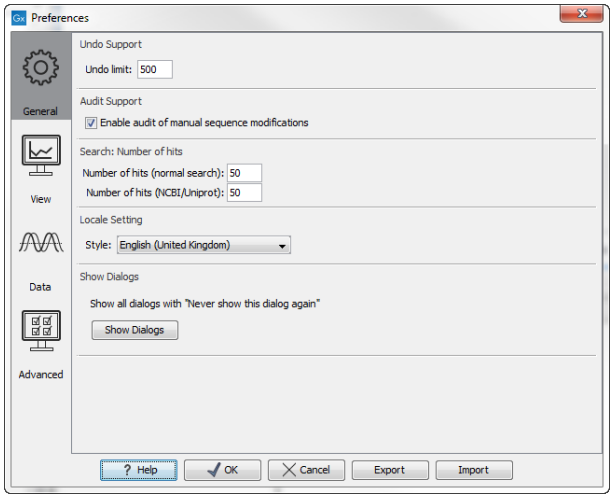


Figure 4.1: *Preferences include General preferences, View preferences, Data preferences, and Advanced settings.*

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

Figure 4.2: *Annotations added when the sequence is edited.*

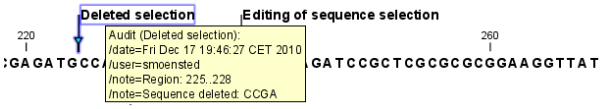


Figure 4.3: *Details of the editing.*

- **Number of hits.** The number of hits shown in *Biomedical Genomics Workbench*, when e.g. searching NCBI. (The sequences shown in the program are not downloaded, until they are opened or dragged/saved into the Navigation Area).
- **Locale Setting.** Specify which country you are located in. This determines how punctation is used in numbers all over the program.
- **Show Dialogs.** A lot of information dialogs have a checkbox: "Never show this dialog again". When you see a dialog and check this box in the dialog, the dialog will not be shown again. If you regret and wish to have the dialog displayed again, click the button in the General Preferences: **Show Dialogs**. Then all the dialogs will be shown again.
- **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).

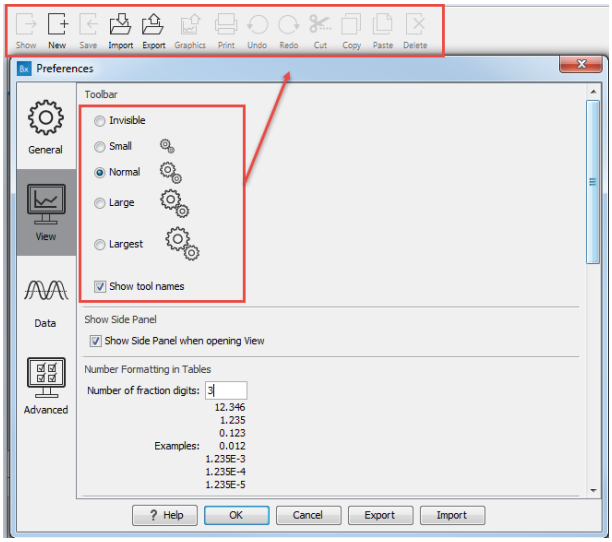


Figure 4.4: *Number formatting of tables.*

2. **Show Side Panel** allows you to choose whether to display the side panel when opening a new view. Note that for any open view, the side panel can be collapsed by clicking on the small triangle at the top left side of the settings area or by using the key combination Ctrl + U (⌘ + U on Mac).

3. **Number formatting in tables** specifies how the numbers should be formatted in tables (see figure 4.5). The examples below the text field are updated when you change the value so that you can see the effect. After you have changed the preference, you have to re-open your tables to see the effect.

4. **Sequence Representation** allows you to change the way the elements appear in the Navigation Area. The following text can be used to describe the element:

   - Name (this is the default information to be shown).

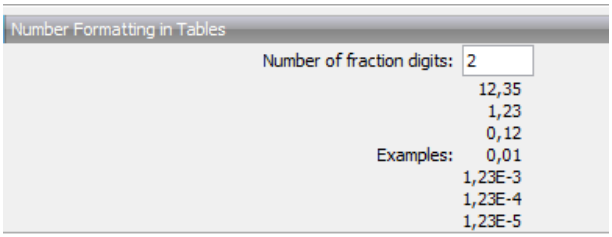


Figure 4.5: *Number formatting of tables.*

- Accession (sequences downloaded from databases like GenBank have an accession number).
- Latin name.
- Latin name (accession).
- Common name.
- Common name (accession).

5. **User Defined View Settings** gives you an overview of the different Side Panel settings that are saved for each view. See section 4.6 to learn more about how to create and save style sheets. If there are other settings beside CLC Standard Settings, you can use this overview to choose which of the settings should be used per default when you open a view (see an example in figure 4.6).

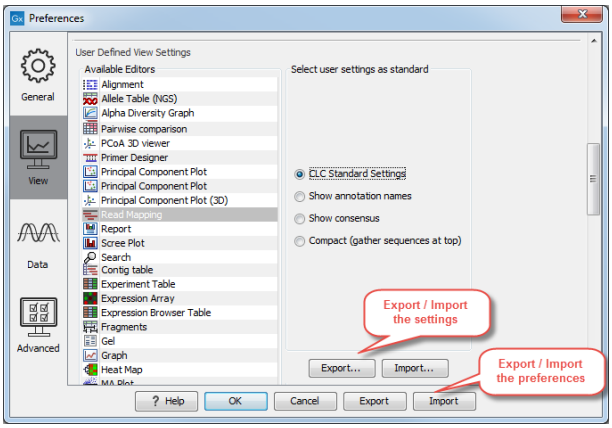


Figure 4.6: *Selecting the default view setting.*

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

6. **Molecule Project 3D Editor** gives you the option to turn off the modern OpenGL rendering for **Molecule Projects** (see section 11.2).

4.2.1 Import and export Side Panel settings

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

To export the **Side Panel** settings, first select the views that you wish to export settings for. Use Ctrl+click (⌘ + click on Mac) or Shift+click to select multiple views. Next click the **Export...** button

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

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

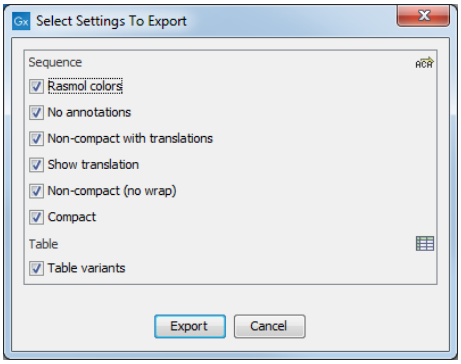


Figure 4.7: *Exporting all settings for circular views.*

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

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

Select the *.vsf file where the settings are saved. The following dialog asks if you wish to overwrite existing **Side Panel** settings, or if you wish to merge the imported settings into the existing ones (see figure 4.8).

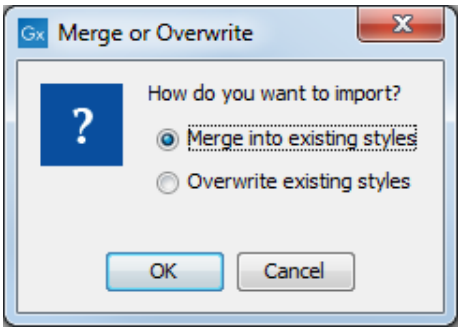


Figure 4.8: *When you import settings, you are asked if you wish to overwrite existing settings or if you wish to merge the new settings into the old ones.*

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

section 6.7).

- 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 32.4.1).
- Linkers for importing 454 data (see section 36.1).
- Linkers for importing Ion Torrent mate pair (see section 6.3.5)

4.4 Advanced preferences

Proxy Settings The Advanced settings include the possibility to set up a proxy server. This is described in section 1.7.

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

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

Read Mapper It is possible to change the size (in MB) of the Read Mapper reference cache.

SRA Download The following options are available:

- **Use Aspera when available** Per default, Aspera is automatically used if installed. This option makes it possible to disable Aspera.
- **Limit Aspera download speeds to [ ] Mb/s (Mac and Linux only)** Using Aspera may take up a lot of network resources. Use this option to specify a maximum download speed (in megabit per second). Note that this option is only available on Mac and Linux. For Windows users, it is possible to limit the maximum download speed by modifying the `aspera.conf file`,

  which can be found in `C:\Program Files (x86)\Aspera\Aspera Connect\etc`. See `http://download.asperasoft.com/download/docs/csrv/3.3.4/linux/html/`

index.html and http://download.asperasoft.com/download/docs/csrv/3.3.4/linux/html/fasp/setting-global-bandwidth.html for more details.

Reference Data It is possible to specify an alternate server URL to use for reference data.

CLC Server Login It is possible to Save the login username and password, initiate automatic server login when opening the workbench, and bypass a proxy server.

4.5 Export/import of preferences

The user preferences of the *Biomedical Genomics Workbench* can be exported to other users of the program, allowing other users to display data with the same preferences as yours. You can also use the export/import preferences function to backup your preferences.

To export preferences, open the **Preferences** dialog and click on the Export bottom at the bottom of the Preferences dialog. Select the relevant preferences and click Export to choose a location to save the exported file(see figure 4.9).

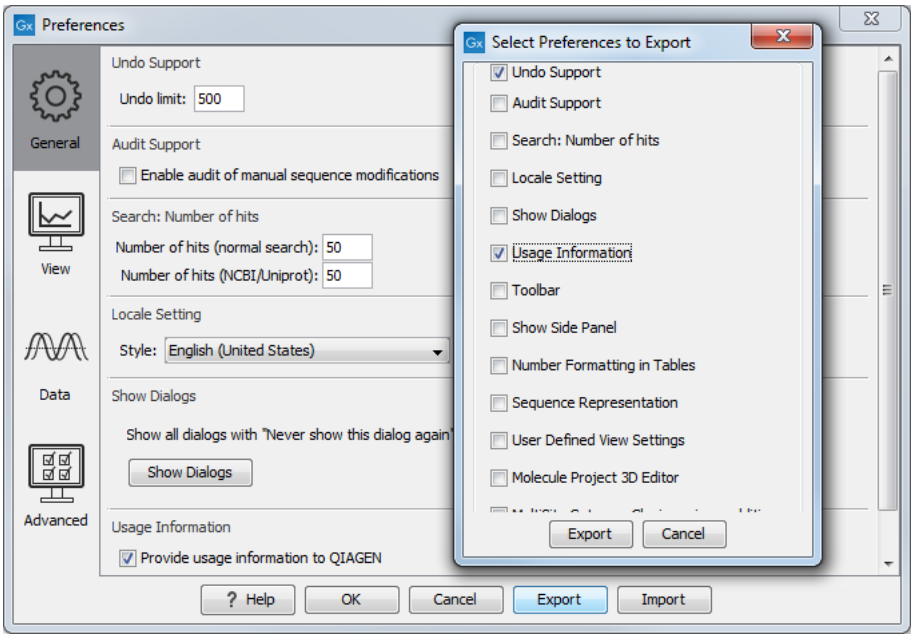


Figure 4.9: *Select which of the preferences you want to export.*

Note! The format of exported preferences is *.cpf. This notation must be submitted to the name of the exported file in order for the exported file to work.

Before exporting, you are asked about which of the different settings you want to include in the exported file. One of the items in the list is "User Defined View Settings". If you export this, only the information about which of the settings is the default setting for each view is exported. If you wish to export the **Side Panel Settings** themselves, see section 4.2.1.

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

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

- Import and export of **bioinformatics data** such as molecules, sequences, alignments etc. (described in section 6.1).
- **Graphics** export of the views that create image files in various formats (described in section 6.7).
- Import and export of **Side Panel Settings** as described in the next section.
- Import and export of all the **Preferences** except the Side Panel settings. This is described above.

4.6 View settings for the Side Panel

The **Side Panel** is shown to the right of all views that are opened in the View Area. Settings are specific to the type of view. Hence, when you save settings of a circular view, they will not be available if you open the sequence in a linear view (see section 2.1.8).

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

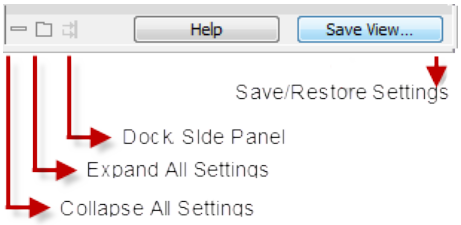


Figure 4.10: *Functionalities found at the bottom of the Side Panel.*

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

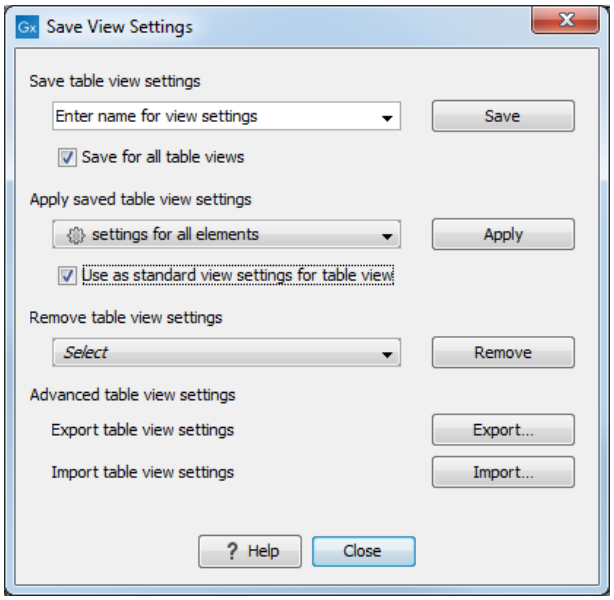


Figure 4.11: *Functionalities found at the bottom of the Side Panel.*

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

- For that view alone, so that the settings will be available to you the next time you open this particular element. The settings are saved with only this element, and will be exported with the element if you later select to export the element to another destination.
- For all other views, when the option "Save for all element views" is checked, so that the settings will be available to you the next time you open any element for which this type of view is available.

Similarly, applying can be done two ways:

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

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

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

Chapter 5

Printing

Contents

Biomedical Genomics Workbench offers different choices of printing the result of your work.

This chapter deals with printing directly from *Biomedical Genomics Workbench*. Another option for using the graphical output of your work, is to export graphics (see chapter 6.7) 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 *Biomedical Genomics Workbench* uses a WYSIWYG principle: What You See Is What You Get. This means that you should use the options in the Side Panel to change how your data, e.g. a sequence, looks on the screen. When you print it, it will look exactly the same way on print as on the screen.

For some of the views, the layout will be slightly changed in order to be printer-friendly.

It is not possible to print elements directly from the **Navigation Area**. They must first be opened in a view in order to be printed. To print the contents of a view:

select relevant view | Print () in the toolbar

This will show a print dialog (see figure 5.1).

In this dialog, you can:

- Select which part of the view you want to print.
- Adjust **Page Setup**.
- See a print **Preview** window.

These three options are described in the three following sections.

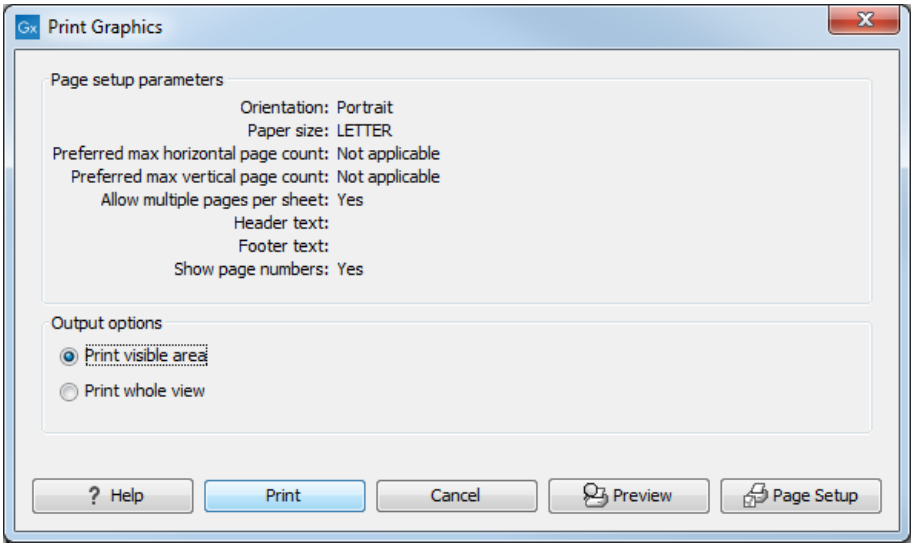


Figure 5.1: *The Print dialog.*

5.1 Selecting which part of the view to print

In the print dialog you can choose to:

- **Print visible area**, or
- **Print whole view**

These options are available for all views that can be zoomed in and out. In figure 5.2 is a view of a circular sequence which is zoomed in so that you can only see a part of it.

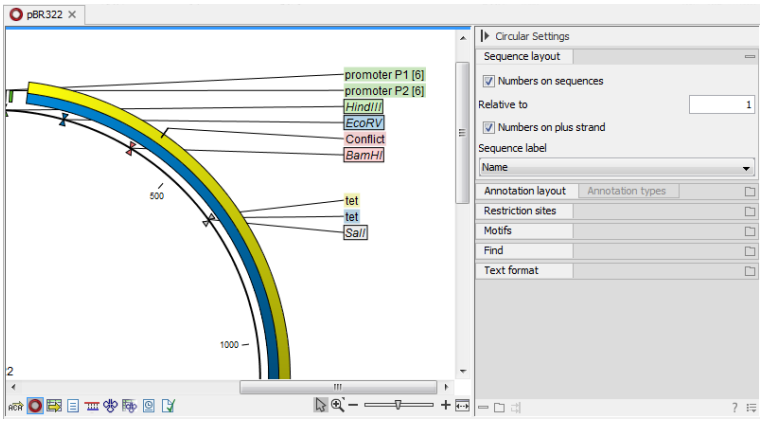


Figure 5.2: *A circular sequence as it looks on the screen.*

When selecting **Print visible area**, your print will reflect the part of the sequence that is *visible* in the view. The result from printing the view from figure 5.2 and choosing **Print visible area** can be seen in figure 5.3.

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

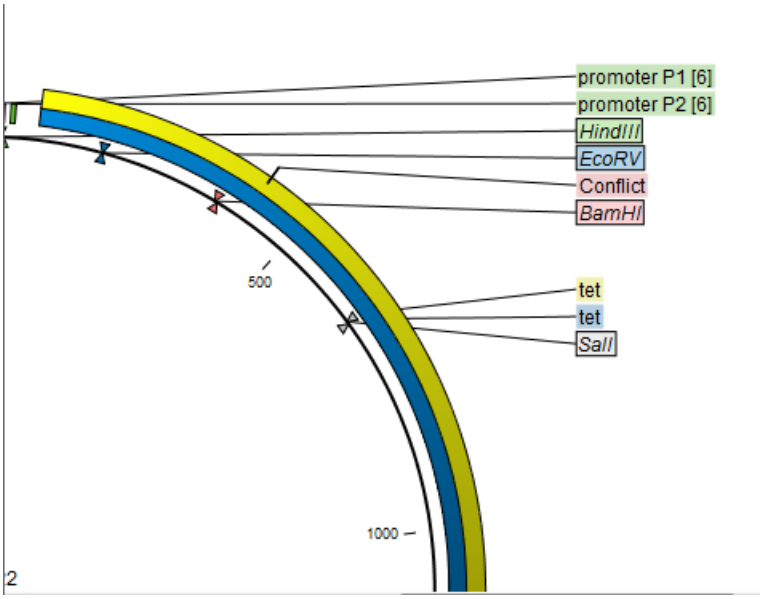


Figure 5.3: *A print of the sequence selecting Print visible area.*

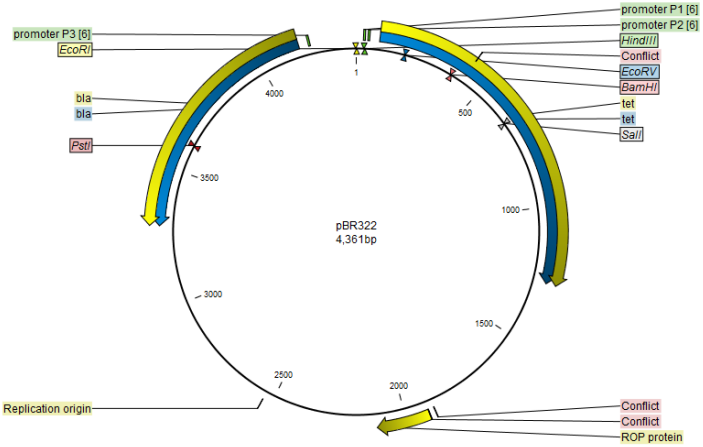


Figure 5.4: *A print of the sequence selecting Print whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*

5.2 Page setup

No matter whether you have chosen to print the visible area or the whole view, you can adjust page setup of the print. An example of this can be seen in figure 5.5

In this dialog you can adjust both the setup of the pages and specify a header and a footer by clicking the tab at the top of the dialog.

You can modify the layout of the page using the following options:

- **Orientation**.
  - **Portrait**. Will print with the paper oriented vertically.
  - **Landscape**. Will print with the paper oriented horizontally.
- **Paper size**. Adjust the size to match the paper in your printer.

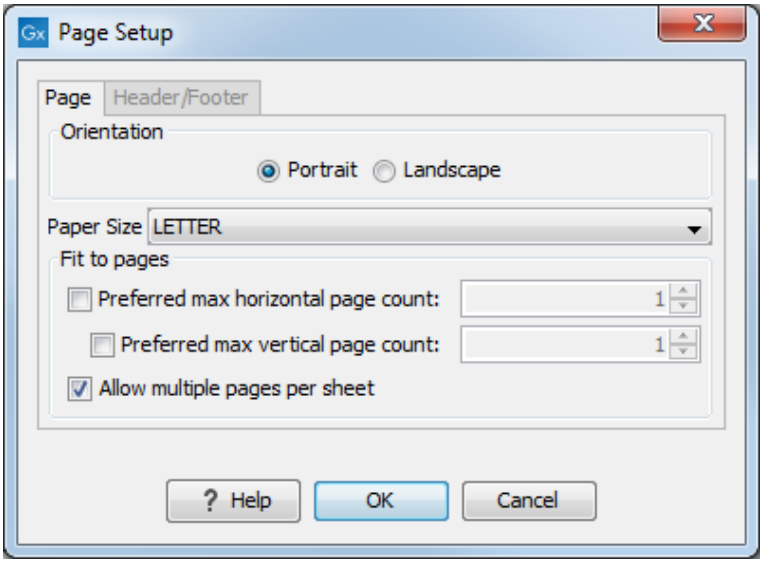


Figure 5.5: *Page Setup.*

- **Fit to pages**. Can be used to control how the graphics should be split across pages (see figure 5.6 for an example).
  - **Horizontal pages**. If you set the value to e.g. 2, the printed content will be broken up horizontally and split across 2 pages. This is useful for sequences that are not wrapped
  - **Vertical pages**. If you set the value to e.g. 2, the printed content will be broken up vertically and split across 2 pages.

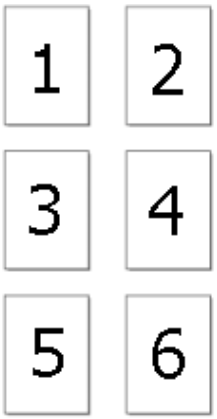


Figure 5.6: *An example where Fit to pages horizontally is set to 2, and Fit to pages vertically is set to 3.*

Note! It is a good idea to consider adjusting view settings (e.g. **Wrap** for sequences), in the **Side Panel** before printing. As explained in the beginning of this chapter, the printed material will look like the view on the screen, and therefore these settings should also be considered when adjusting **Page Setup**.

Header and footer Click the **Header/Footer** tab to edit the header and footer text. By clicking in the text field for either **Custom header text** or **Custom footer text** you can access the auto

formats for header/footer text in **Insert a caret position**. Click either **Date**, **View name**, or **User name** to include the auto format in the header/footer text.

Click **OK** when you have adjusted the **Page Setup**. The settings are saved so that you do not have to adjust them again next time you print. You can also change the **Page Setup** from the **File** menu.

5.3 Print preview

The preview is shown in figure 5.7.

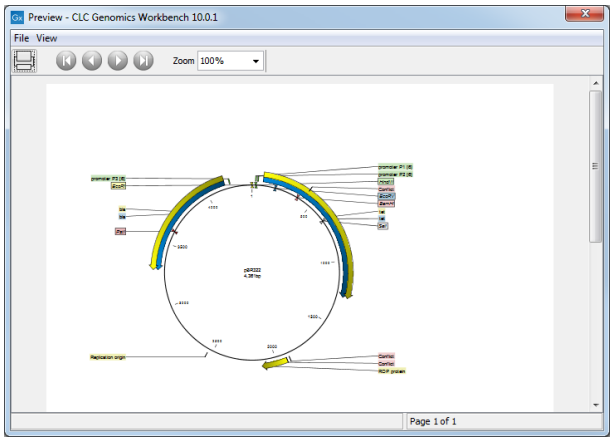


Figure 5.7: *Print preview.*

The **Print preview** window lets you see the layout of the pages that are printed. Use the arrows in the toolbar to navigate between the pages. Click Print ( ) to show the print dialog, which lets you choose e.g. which pages to print.

The **Print preview** window is for preview only - the layout of the pages must be adjusted in the **Page setup**.

Chapter 6

Import/export of data and graphics

Contents

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

This chapter first deals with importing and exporting data in bioinformatic data formats and as external files. Next comes an explanation of how to export graph data points to a file, and how to export graphics.

For **import of NGS data**, please see section 6.3.

Using data from other workbenches Please note that if you also have access to CLC Genomics Workbench, CLC Main Workbench, or CLC Sequence Viewer you may have generated different types of output that you would like to view in the *Biomedical Genomics Workbench*. All types of output that have been created in CLC Genomics Workbench, CLC Main Workbench, or CLC Sequence Viewer can be opened in the *Biomedical Genomics Workbench*. This means that you are capable of opening certain output types that cannot be generated from within the *Biomedical Genomics Workbench*. In such cases we refer to our other manuals (`http://www.qiagenbioinformatics.com/support/manuals/`) for further information about the output types that are not described in the *Biomedical Genomics Workbench* manual.

Output files from other workbenches can be imported as described in section 6.1.

6.1 Standard import

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

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

For **import of NGS data**, please see section 6.3 For import of tracks, please see section 6.2.

Import using the import dialog To start the import using the import dialog:

click Import () in the Toolbar | Standard Import

This will show a dialog similar to figure 6.1. You can change which kind of file types that should be shown by selecting a file format in the **Files of type** box.

Next, select one or more files or folders to import and click **Next** to select a place for saving the result files. If you import one or more folders, the contents of the folder is automatically imported and placed in that folder in the Navigation Area. If the folder contains subfolders, the whole folder structure is imported.

In the import dialog (figure 6.1), there are three import options:

Automatic import This will import the file and *Biomedical Genomics Workbench* will try to determine the format of the file. The format is determined based on the file extension (e.g.

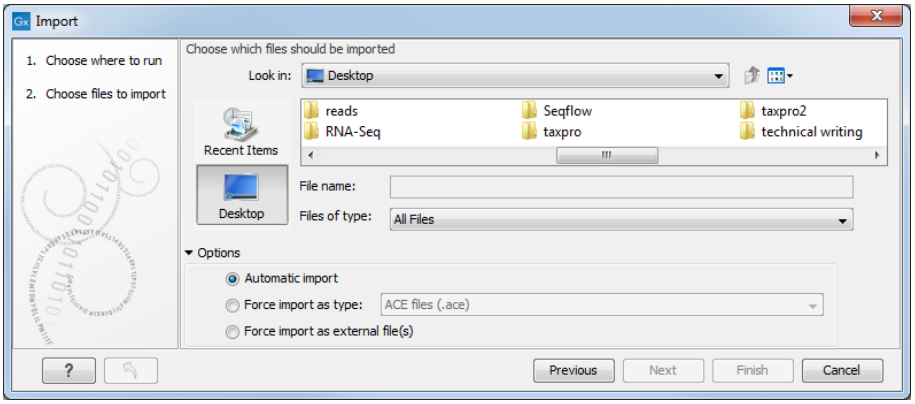


Figure 6.1: *The import dialog.*

SwissProt files have .swp at the end of the file name) in combination with a detection of elements in the file that are specific to the individual file formats. If the file type is not recognized, it will be imported as an external file. In most cases, automatic import will yield a successful result, but if the import goes wrong, the next option can be helpful:

Force import as type This option should be used if *Biomedical Genomics Workbench* cannot successfully determine the file format. By forcing the import as a specific type, the automatic determination of the file format is bypassed, and the file is imported as the type specified.

Force import as external file This option should be used if a file is imported as a bioinformatics file when it should just have been external file. It could be an ordinary text file which is imported as a sequence.

Import using drag and drop It is also possible to drag a file from e.g. the desktop into the **Navigation Area** of *Biomedical Genomics Workbench*. This is equivalent to importing the file using the **Automatic import** option described above. If the file type is not recognized, it will be imported as an external file.

Import using copy/paste of text If you have e.g. a text file or a browser displaying a sequence in one of the formats that can be imported by *Biomedical Genomics Workbench*, there is a very easy way to get this sequence into the **Navigation Area**:

Copy the text from the text file or browser | Select a folder in the Navigation Area | Paste ( )

This will create a new sequence based on the text copied. This operation is equivalent to saving the text in a text file and importing it into the *Biomedical Genomics Workbench*.

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

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

6.1.1 External files

In order to help you organize your research projects, *Biomedical Genomics Workbench* lets you import all kinds of files. E.g. if you have Word, Excel or pdf-files related to your project, you can import them into the **Navigation Area** of *Biomedical Genomics Workbench*. Importing an external file creates a copy of the file which is stored at the location you have chosen for import. The file can now be opened by double-clicking the file in the **Navigation Area**. The file is opened using the default application for this file type (e.g. Microsoft Word for .doc-files and Adobe Reader for .pdf).

External files are imported and exported in the same way as bioinformatics files (see section 6.1). Bioinformatics files not recognized by *Biomedical Genomics Workbench* are also treated as external files.

6.2 Import tracks

Tracks (see chapter 19) are imported in a special way, because extra information is needed in order to interpret the files correctly.

Tracks are imported using: **click Import ( ) in the Toolbar | Tracks**
This will open a dialog as shown in figure 6.2.

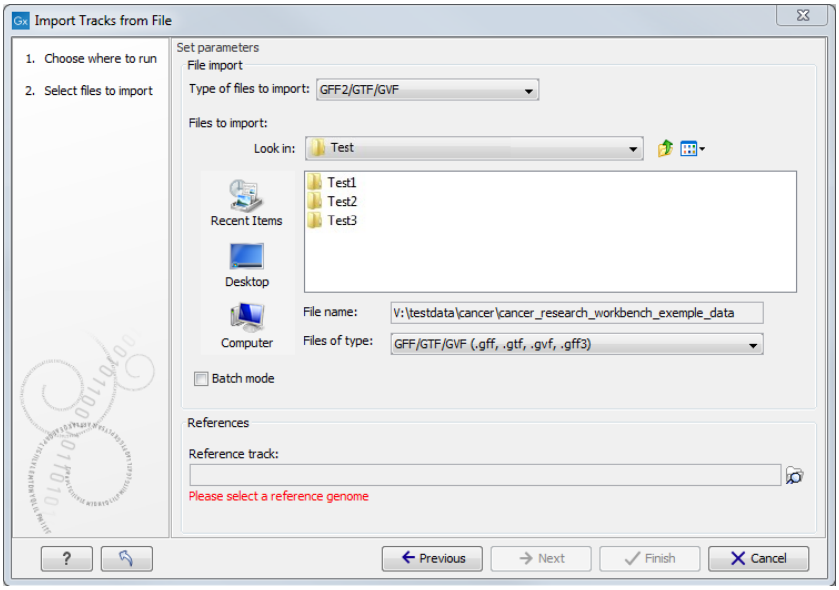


Figure 6.2: *Select files to import.*

At the top, you select the file type to import. Below, select the files to import. If import is performed with the batch option selected, then each file is processed independently and separate tracks are produced for each file. If the batch option is not selected, then variants for all files will be added to the same track (or tracks in the case VCF files including genotype information). The formats currently accepted are:

FASTA This is the standard fasta importer that will produce a sequence track rather than a standard fasta sequence.

GFF2/GTF/GVF A GFF2/GTF file does not contain any sequence information, it only contains a list of various types of annotations. A GVF file is similar to a GFF file but uses Sequence

Ontology to describe genome variation data. For these formats, the importer adds the annotation in each of the lines in the file to the chosen sequence, at the position or region in which the file specifies that it should go, and with the annotation type, name, description etc. as given in the file. However, 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 are 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). For a comprehensive source of genomic annotation of genes and transcripts, we refer to the Ensembl web site at http://www.ensembl.org/info/data/ftp/index.html. On this page, you can download GTF files that can be used to annotate genomes for use in other analyses in the workbench. You can also read more about these formats at http://www.sanger.ac.uk/resources/software/gff/spec.html, http://mblab.wustl.edu/GTF22.html and https://genomebiology.biomedcentral.com/articles/10.1186/gb-2010-11-8-r88.

GFF3 A GFF3 file contains a list of various types of annotations that can be linked together with "Parent" and "ID" tags. Learn more about how the workbench handles GFF3 format in section 6.2.1.

VCF This is the file format used for variants by the 1000 Genomes Project and it has become a standard format. Read how to access data at http://www.1000genomes.org/data#DataAccess. When importing a single VCF file, you will get a track for each sample contained in the VCF file. In cases where more than one sample is contained in a VCF file, you can choose to import the files together or individually by using the batch mode found in the lower left side of the wizard shown in figure 6.2. The difference between the two import modes is that the batch mode will import the samples individually in separate track files, whereas the non-batch mode will keep variants for one sample in one track, thus merging samples from the different input files (in cases where the same sample is contained in different input files). If you import more than one VCF file that each contain more than one sample, the non-batch mode will generate one track file for each unique sample. The batch mode will generate a track file for each of the original VCF files with the entire content, as if importing each of the VCF files one by one. E.g. VCF file 1 contains sample 1 and sample 2, and VCF file 2 contains sample 2 and sample 3. When VCF file 1 and VCF file 2 are imported in non-batch mode, you will get three individual track files; one for each of the three samples 1, 2, and 3. If VCF file 1 and VCF file 2 were instead imported using

the batch function, the result of the import would be four track files: a track from sample 1 from file 1, a track from sample 2 from file 1, a track from sample 2 from file 2, and a track from sample 3 from file 2.

Complete Genomics master var file This is the file format used by Complete Genomics for all kinds of variant data and can be used to analyze and visualize the variant calls made by Complete Genomics. Please note that you can import evidence files with the read alignments into the *CLC Genomics Workbench* as well (refer to the Complete Genomics import section of the Workbench user manual).

BED Simple format for annotations. Read more at http://genome.ucsc.edu/FAQ/FAQformat.html#format1. This format is typically used for very simple annotations, for example target regions for sequence capture methods.

Wiggle The Wiggle format as defined by UCSC (http://genome.ucsc.edu/goldenPath/help/wiggle.html), is used to hold continuous data like conservation scores, GC content etc. When imported into the *Biomedical Genomics Workbench*, a graph track is created. An example of a popular Wiggle file is the conservation scores from UCSC which can be download for human from http://hgdownload.cse.ucsc.edu/goldenPath/hg19/phastCons46way/.

UCSC variant database table dump Table dumps of **variant** annotations from the UCSC can be imported using this option. Mainly files ending with `.txt.gz` on this list can be used: http://hgdownload.cse.ucsc.edu/goldenPath/hg19/database/. Please note that importer is for variant data and is not a general importer for all annotation types. This is mainly intended to allow you to import the popular *Common SNPs* variant set from UCSC. The file can be downloaded from the UCSC web site here: http://hgdownload.cse.ucsc.edu/goldenPath/hg19/database/snp138Common.txt.gz. Other sets of variant annotation can also be downloaded in this format using the UCSC Table Browser.

COSMIC variation database This lets you import the COSMIC database, which is a well-known publicly available primary database on somatic mutations in human cancer. The file can be downloaded from the UCSC web site here: http://cancer.sanger.ac.uk/cancergenome/projects/cosmic/download, Users must first register to download the database. Import the file as a track. Through Import->Tracks we support certain COSMIC databases in tsv format that can be manually downloaded from the COSMIC ftp site:

- COSMIC Complete mutation data: CosmicCompleteTargetedScreensMutantExport.tsv
- COSMIC Mutation Data (Genome Screens): CosmicGenomeScreensMutantExport.tsv
- COSMIC Mutation Data : CosmicMutantExport.tsv
- All Mutations in Census Genes : CosmicMutantExportCensus.tsv

Please see chapter H.1.7 for more information on how different formats (e.g. VCF and GVF) are interpreted during import in CLC format. For all of the above, zip files are also supported. Please note that for human data, there is a difference between the UCSC genome build and Ensembl/NCBI for the mitochondrial genome. This means that for the mitochondrial genome, data from UCSC should not be mixed with data from other sources (see http://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/). Most of the data above is annotation data and if the file includes information about allele variants (like VCF, Complete Genomics and GVF), it will be combined into one **variant** track that can be used for finding known variants in your

experimental data. When the data cannot be recognized as variant data, one track is created for each annotation type. Genome / gene annotation tracks can be automatically imported from relevant databases as described here: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Selecting_data_types_download.html. For all types of files except fasta, you need to select a reference track as well. This is because most the annotation files do not contain enough information about chromosome names and lengths which are necessary to create the appropriate data structures.

6.2.1 GFF3 format

A GFF3 file contains a list of various types of annotations that can be linked together with "Parent" and "ID" tags.

Here are some example of a few common tags used by the format:

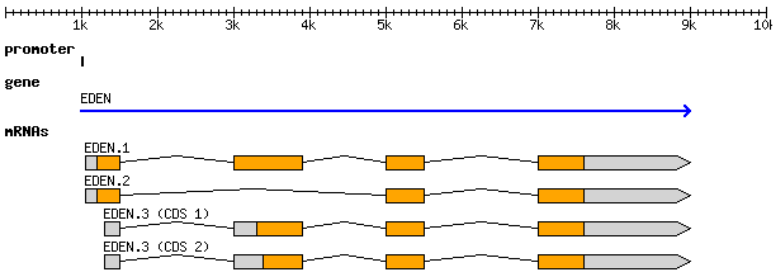
- **ID** IDs for each feature must be unique within the scope of the GFF file. In the case of discontinuous features (i.e., a single feature that exists over multiple genomic locations) the same ID may appear on multiple lines. All lines that share an ID collectively represent a single feature.
- **Parent** A parent ID can be used to group exons into transcripts, transcripts into genes, and so forth. A feature may have multiple parents. A parent ID can only be used to indicate a 'part of' relationship.
- **Name** The name that will be displayed as a label in the track view. Unlike IDs, there is no requirement that the Name be unique within the file.

Figure 6.3 exemplifies how tags are used to create annotations.

In the workbench, the GFF3 importer will create an output track for each feature type present in the file.

- **Gene-like types**. These are types described in the Sequence Ontology as being subtypes of genes, e.g. ncRNA_gene, plastid_gene, tRNA_gene. Gene-like types are gathered together into an aggregated track with a name of the form "myFileName (Gene)". We recommend that users use this file in RNA-Seq.
- **Transcript-like types**. These are types described in the Sequence Ontology as being subtypes of transcripts that are neither primary transcripts (i.e., they do not require further processing to become functional), nor fusion transcripts. Again, there are several dozen, such as mRNA, lnc_RNA, threonyl_RNA. Transcript-like types are gathered together into an aggregated track with a name of the form "myFileName (RNA)". We recommend that users use this file in RNA-Seq.
- **Exons**. Where possible, exons are merged into their parent features. For example, the output of the lines shown in figure 6.4 will be a single mRNA feature with four exonic regions (from 1300 to 1500, 3000 to 3902, 5000 to 5500,and 7000 to 9000), and no exon features will be output on their own.

  However, in cases where the parent is of a "gene-like" type, exons are output as their own independent features in the exon track. Finding a lot of features in the exon track




```
0  ##gff-version 3.2.1
1  ##sequence-region ctg123 1 1497228
2  ctg123 . gene            1000  9000  .  +  .  ID=gene00001;Name=EDEN
3  ctg123 . TF_binding_site 1000  1012  .  +  .  ID=tfbs00001;Parent=gene00001
4  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00001;Parent=gene00001;Name=EDEN.1
5  ctg123 . mRNA            1050  9000  .  +  .  ID=mRNA00002;Parent=gene00001;Name=EDEN.2
6  ctg123 . mRNA            1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001;Name=EDEN.3
7  ctg123 . exon            1300  1500  .  +  .  ID=exon00001;Parent=mRNA00003
8  ctg123 . exon            1050  1500  .  +  .  ID=exon00002;Parent=mRNA00001,mRNA00002
9  ctg123 . exon            3000  3902  .  +  .  ID=exon00003;Parent=mRNA00001,mRNA00003
10 ctg123 . exon            5000  5500  .  +  .  ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11 ctg123 . exon            7000  9000  .  +  .  ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12 ctg123 . CDS             1201  1500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13 ctg123 . CDS             3000  3902  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14 ctg123 . CDS             5000  5500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15 ctg123 . CDS             7000  7600  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16 ctg123 . CDS             1201  1500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17 ctg123 . CDS             5000  5500  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18 ctg123 . CDS             7000  7600  .  +  0  ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19 ctg123 . CDS             3301  3902  .  +  0  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20 ctg123 . CDS             5000  5500  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21 ctg123 . CDS             7000  7600  .  +  1  ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22 ctg123 . CDS             3391  3902  .  +  0  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23 ctg123 . CDS             5000  5500  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24 ctg123 . CDS             7000  7600  .  +  1  ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```

Figure 6.3: *Example of a GFF3 file and the corresponding annotations from https://github.com/The-Sequence-Ontology/Specifications/blob/master/gff3.md.*

```
ctg123 . mRNA            1300 9000  .  +  .  ID=mRNA00003;Parent=gene00001
ctg123 . exon            1300 1500  .  +  .  Parent=mRNA00003
ctg123 . exon            3000 3902  .  +  .  Parent=mRNA00003
ctg123 . exon            5000 5500  .  +  .  Parent=mRNA00003
ctg123 . exon            7000 9000  .  +  .  Parent=mRNA00003
```

Figure 6.4: *Exons will be merged into their parent features when the parent is not a "gene-like" type.*

can suggest a problem with the file being imported. However, with large databases, this is more likely to be due to the database creators choosing to represent pseudogenes as exons with no transcript.

- **CDS** CDS regions with the same parent are joined together into a single spliced feature. If CDS features do not have a parent they are instead joined based on their ID, as for any other feature (described below)

- **Features with the same ID** Regardless of the feature type, features that have the same ID are merged into a single spliced feature. For example, the output of the following figure 6.5 will be a single cDNA_match feature with regions (1050..1500, 5000..5500, 7000..9000).

Naming of features

When one of the following qualifiers is present, it will be used for naming in the prioritized order:

```
ctg123 . cDNA_match 1050  1500  5.8e-42 +  . ID=match00001;Target=cdna0123 12  462
ctg123 . cDNA_match 5000  5500  8.1e-43 +  . ID=match00001;Target=cdna0123 463 963
ctg123 . cDNA_match 7000  9000  1.4e-40 +  . ID=match00001;Target=cdna0123 964 2964
```

Figure 6.5: *Features that have the same ID are merged into a single spliced feature.*

1. the "Name" of the feature
2. the "Name" of the first named parent of the feature
3. the "ID" of the feature
4. the "ID" of the first parent
5. the type of the feature

Several examples of naming strategies are depicted in figure 6.6.

1) The mRNA from the following will be called "EDEN-001"

```
ctg123 . gene           1000  9000  .  +  .  ID=gene00001;Name=EDEN
ctg123 . mRNA           1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001,Name=EDEN-001
```

2) The mRNA from the following will be called "EDEN"

```
ctg123 . gene           1000  9000  .  +  .  ID=gene00001;Name=EDEN
ctg123 . mRNA           1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001
```

3) The mRNA from the following will be called "mRNA00003"

```
ctg123 . gene           1000  9000  .  +  .  ID=gene00001
ctg123 . mRNA           1300  9000  .  +  .  ID=mRNA00003;Parent=gene00001
```

4) The mRNA from the following will be called "gene00001"

```
ctg123 . gene           1000  9000  .  +  .  ID=gene00001
ctg123 . mRNA           1300  9000  .  +  .  Parent=gene00001
```

5) The mRNA from the following mRNA without a parent gene will be called "mRNA"

```
ctg123 . mRNA           1300  9000  .  +  .
```

Figure 6.6: *Naming of features.*

Merged CDS features have a slightly different naming scheme. First, if a CDS feature in the GFF3 file has more than one parent, we create one CDS feature in the workbench for each parent, and each is merged with all other CDS features from the GFF3 file that has the parent feature as parent as well. The naming is then done in the following prioritized order:

1. the "Name" of the feature, if all the constituent CDS features have the same "Name".
2. the "Name" of the first named parent of the feature, if it has a name.
3. the "Name" of the first of the merged CDS features with a name.
4. the "ID" of the first of the merged CDS features with an ID.
5. the "ID" of the parent.

For features with the same ID, the naming scheme is as follows:

1. the "Name" of the feature, if all have the same "Name".

2. If there is a set of common parents for the features and one of the common parents have a "Name", the name of the first common parent with a "Name" is used.
3. If at least one feature has a name, the name of the first feature with the name is used.
4. the "ID" of the first of the features

Limits of the GFF3 importer

Features are imported only if their SeqID (i.e., the value in the first column of the gff3) can be matched to the name of a chromosome in the genome. Matching need not be exact (see section H.1.7). However, in some cases it may be necessary to manually edit either the names of the genomic sequences (for example in a fasta file), or the SeqIDs in the GFF3 file so that they match. Features without a match aren't imported. You can see the number of skipped features in the importer log.

The start and stop position of a feature cannot extend beyond the ends of a chromosome, unless the chromosome is explicitly marked as circular, which is indicated by << and >> at the beginning and the end of the sequence.

Trying to import such a file will fail. One option is to delete the feature that extends beyond the end of the chromosome and to start the import again.

The following instances are not supported:

- Interpreting SOFA accession numbers. The type of the feature is constrained to be either: (a) a term from the "lite" sequence ontology, SOFA; or (b) a SOFA accession number, distinguished using the syntax SO:000000. The importer recognizes terms from SOFA as well as terms from the full Sequence Ontology, but will not translate accession numbers to types. So for example, features with type SO:0000316 will not be interpreted as "CDS" but will be handled like any other type.
- The fasta directive ##FASTA. This FASTA section situated at the end of a GFF3 file specifies sequences of ESTs as well as of contigs. The GFF3 importer will ignore these sequences.
- Alignments. An aligned feature is handled as a single region, with the Gap and Target attributes added as annotations. We do not use Gap and Target to show how the feature aligns.
- Comment lines. We do not interpret lines beginning with a #. Especially relevant are lines "##sequence-region seqid start end" which some parsers use to perform bounds checking of features. Our bounds checking is instead performed against the user-supplied genome.

6.3 Import high-throughput sequencing data

The *Biomedical Genomics Workbench* has dedicated tools for importing data from the following High-throughput sequencing systems.

- Illumina's Genome Analyzer, Nextseq, HiSeq and MiSeq
- PacBio

- Ion Torrent
- Complete Genomics (only processed data - master var and evidence files)

Importers for Roche 454 and SOLiD are also available in the Legacy Tools folder.

The reason for having dedicated tools for this is to standardize the data so that most downstream analyses and visualization of the data works seamlessly with all sequencing platforms. In case a sequence list was not imported with the right tool, it is possible to edit "Read Group" information in the "Element Info" view: choose from the drop-down menu the sequencing platform that was used to generate the data (figure 6.7) and click OK.

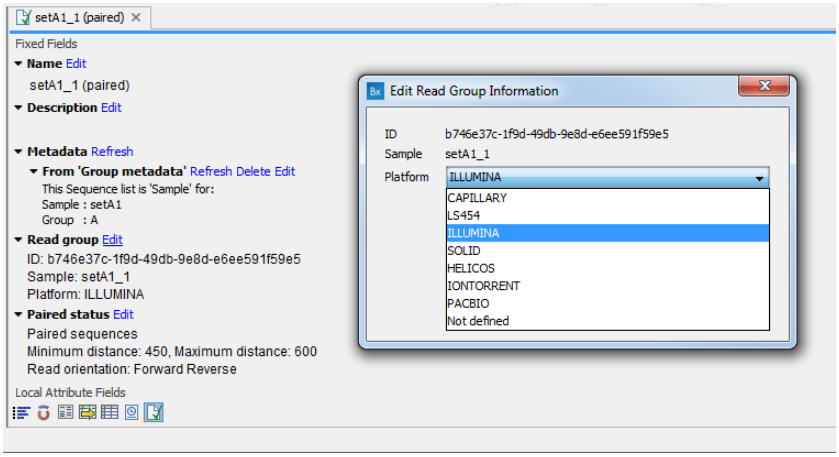


Figure 6.7: *Editing the platform that was used to generate the data in the "Element Info" view.*

In addition to these formats, mapped data in SAM/BAM format can also be imported.

Clicking on the **Import** ( ) button in the top toolbar will bring up a list of the supported data types as shown in figure 6.8.

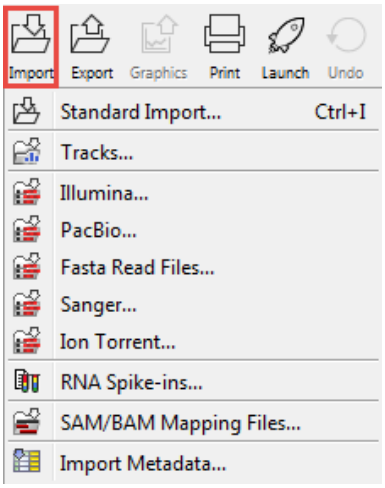


Figure 6.8: *Choosing what kind of data you wish to import.*

Select the appropriate format and then fill in the information as explained in the following sections.

Please note that alignments of *Complete Genomics* data can be imported using the SAM/BAM importer, see section 6.3.6 below.

6.3.1 Illumina

The *Biomedical Genomics Workbench* supports data from Illumina's Genome Analyzer, HiSeq 2000, NextSeq and the MiSeq systems. Choosing the Illumina import will open the dialog shown in figure 6.9.

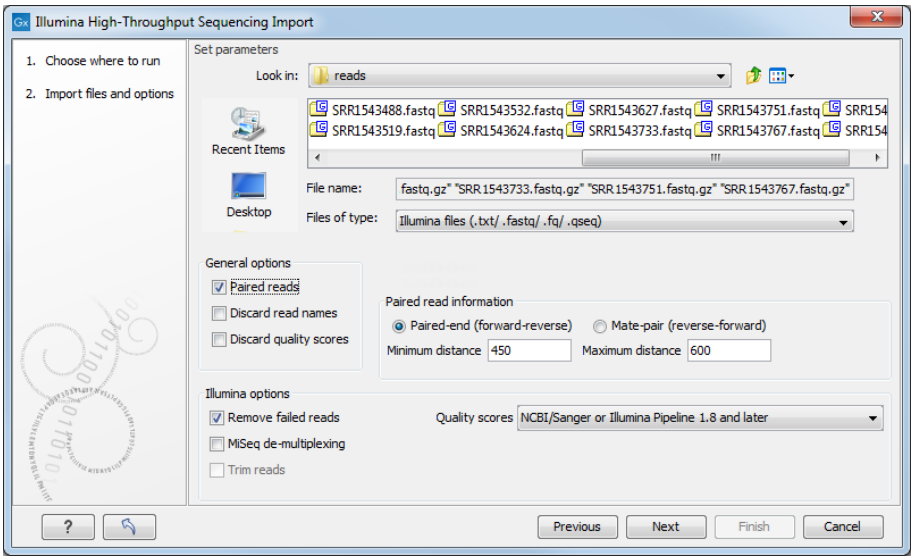


Figure 6.9: *Importing data from Illumina systems.*

File format The file formats accepted are:

- Fastq
- Scarf
- Qseq
- For all formats, compressed data in gzip format is also supported (.gz).

Paired data in any of these formats can be imported.

Note that there is information inside qseq and fastq files specifying whether a read has passed a quality filter or not. If you check **Remove failed reads** these reads will be ignored during import. For qseq files there is a flag at the end of each read with values 0 (failed) or 1 (passed). In this example, the read is marked as failed and if Remove failed reads is checked, the read is removed.

```
M10  68  1  1  28680  29475  0  1  CATGGCCGTACAGGAAACACACATCATAGCATCACACGA  BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB  0
```

For fastq files, part of the header information for the quality score has a flag where Y means failed and N means passed. In this example, the read has not passed the quality filter:

```
@EAS139:136:FC706VJ:2:2104:15343:197393 1:Y:18:ATCACG
```

Note! In the **Illumina pipeline 1.5-1.7**, the letter B in the quality score has a special meaning. 'B' is used as a trim clipping. This means that when selecting Illumina pipeline 1.5-1.7, the *reads are automatically trimmed* when a B is encountered at either end of the reads in the input file. This will happen also if you choose to discard quality scores during import.

General Options The **General options** to the left are:

- **Paired reads**. For paired import, you can select whether the data is **Paired-end** or **Mate-pair**. For paired data, the Workbench expects the first reads of the pairs to be in one file and the second reads of the pairs to be in another. So, for example, if you had specified that the pairs were in forward-reverse orientation, then the first file would be assumed to contain the forward reads. The second file would be assumed to contain the reverse reads.

  When loading files containing paired data, the *Biomedical Genomics Workbench* sorts the files selected according to rules based on the file naming scheme:

  – For files coming off the CASAVA1.8 pipeline, we organize pairs according to their identifier and chunk number. Files named with `_R1_` are assumed to contain the first sequences of the pairs, and those with `_R2_` in the name are assumed to contain the second sequence of the pairs.

  – For other files, we sort them all alphanumerically, and then group them two by two. This means that files 1 and 2 in the list are loaded as pairs, files 3 and 4 in the list are seen as pairs, and so on.

  In the simplest case, the files are typically named as shown in figure 6.9. In this case, the data is paired end, and the file containing the forward reads is called `s_1_1_sequence.txt` and the file containing reverse reads is called `s_1_2_sequence.txt`. Other common filenames for paired data, like `_1_sequence.txt`, `_1_qseq.txt`, `_2_sequence.txt` or `_2_qseq.txt` will be sorted alphanumerically. In such cases, files containing the final `_1` should contain the first reads of a pair, and those containing the final `_2` should contain the second reads of a pair.

  For files from CASAVA1.8, files with base names like these: ID_R1_001, ID_R1_002, ID_R2_001, ID_R2_002 would be sorted in this order:

  1. ID_R1_001
  2. ID_R2_001
  3. ID_R1_002
  4. ID_R2_002

  The data in files ID_R1_001 and ID_R2_001 would be loaded as a pair, and ID_R1_002, ID_R2_002 would be loaded as a pair.

  Within each file, the first read of a pair will have a `1` somewhere in the information line. In most cases, this will be a `/1` at the end of the read name. In some cases though (e.g. CASAVA1.8), there will be a `1` elsewhere in the information line for each sequence. Similarly, the second read of a pair will have a `2` somewhere in the information line - either a `/2` at the end of the read name, or a `2` elsewhere in the information line.

  If you do not choose to discard your read names on import (see next parameter setting), you can quickly check that your paired data has imported in the pairs you expect by looking at the first few sequence names in your imported paired data object. The first two sequences should have the same name, except for a `1` or a `2` somewhere in the read name line.

  Paired-end and mate-pair data are handled the same way with regards to sorting on filenames. Their data structure is the same the same once imported into the Workbench. The only difference is that the expected orientation of the reads: reverse-forward in the

case of mate pairs, and forward-reverse in the case of paired end data. Read more about handling paired data in section 6.3.7.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard read names to save disk space.
- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. Read more about the quality scores of Illumina below.

Paired read information These options become available if you selected the option "Paired reads" in the General options. First, it is very important to select the correct nature of the paired reads imported: Paired-end (forward-reverse) or Mate-pair (reverse-forward). Second, you have to specify the Minimum and Maximum distances for your pairs. The paired read distance includes the full read sequence, which means that is from the beginning of the forward read to the beginning of the reverse read (figure 6.10). The distances are usually defined during the library preparation of your sequencing experiment, but in doubt you can enter default values: for paired-end the distances distances are between 1 and 1000 bp while mate-pair reads typically have longer distances between 1000-5000 bp (and sometimes up to 10000). Note that the tools usually used subsequently to process Illumina reads (such as "Map Reads to Reference" or "RNA-Seq Analysis") have an "Auto-detect paired distances" option that is enabled by default. As long as this option is used, mis-specifying the distances during import should bear no consequences.

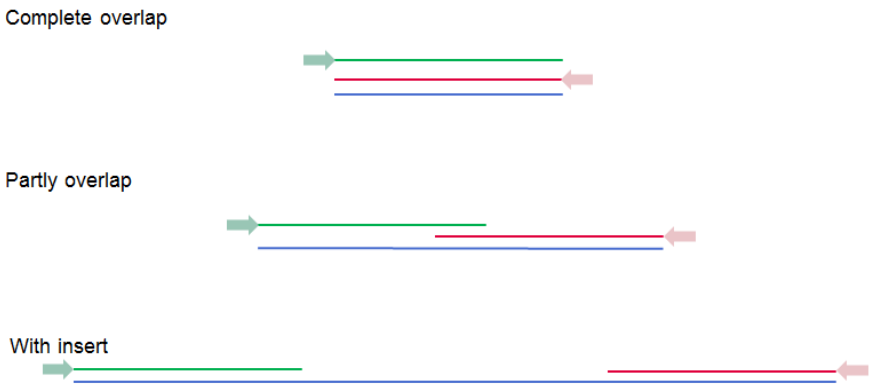


Figure 6.10: *Green lines represent forward reads, red lines reverse reads, and in blue is shown the distance of the sequenced DNA fragment. Thus, if there is a complete overlap, the minimum distance will not be 0, but the length of the overlap.*

Illumina options

- **Remove failed reads**. If you check **Remove failed reads**, reads that did not pass a quality filter (in qseq and fastq files) will be ignored during import. For more information on format specific quality filters see section on file format above). If you import paired data and one

read in a pair is removed during import, the remaining mate will be saved in a separate sequence list with single reads.

- **MiSeq de-multiplexing**. Using this option on MiSeq multiplexed data will divide reads into different files based on the "IndexSequence" of the read header:

```
@Instrument:RunID:FlowCellID:Lane:Tile:X:Y:UMI ReadNum:FilterFlag:0:IndexSequence
```

  Subsequent analysis can then be executed in batch on all the files, and results can be compared at the end.

- **Trim reads**. This option applies to Illumina Pipeline 1.5 to 1.7. In this pipeline, the value 2 (B) has special meaning and is used as a trim clipping. This means that when selecting Illumina Pipeline 1.5 and later, the reads are trimmed when a B is encountered at either end of the reads in the input file if the **Trim reads** option is checked.

Click **Next** to adjust how to handle the results (see section 8.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 8.3).

Quality scores in the Illumina platform

The quality scores in the FASTQ format come in different versions. You can read more about the FASTQ format at http://en.wikipedia.org/wiki/FASTQ_format. When you select to import Illumina data and click **Next** there is an option to use different quality score schemes at the bottom of the dialog (see figure 6.11).

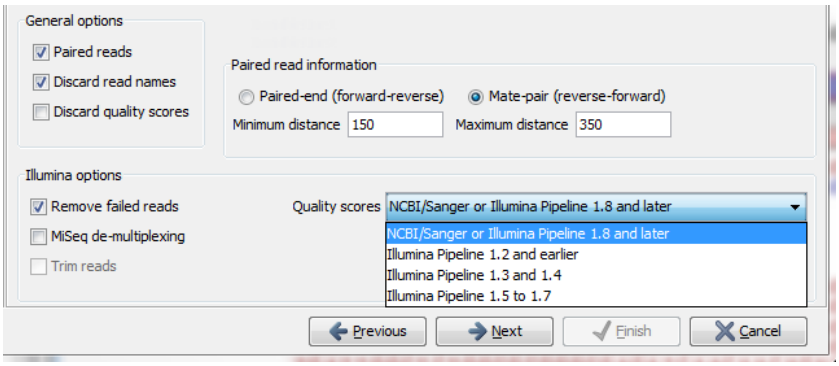


Figure 6.11: *Selecting the quality score scheme.*

There are four options:

- **NCBI/Sanger or Illumina 1.8 and later**. Using a Phred scale encoded using ASCII 33 to 93. This is the standard for fastq formats except for the early Illumina data formats (this changed with version 1.8 of the Illumina Pipeline).

- **Illumina Pipeline 1.2 and earlier**. Using a Solexa/Illumina scale (-5 to 40) using ASCII 59 to 104. The Workbench automatically converts these quality scores to the Phred scale on import in order to ensure a common scale for analyses across data sets from different platforms (see details on the conversion next to the sample below).

- **Illumina Pipeline 1.3 and 1.4**. Using a Phred scale using ASCII 64 to 104.
- **Illumina Pipeline 1.5 to 1.7**. Using a Phred scale using ASCII 64 to 104. Values 0 (@) and 1 (A) are not used anymore. Value 2 (B) has special meaning and is used as a trim clipping. This means that when selecting Illumina Pipeline 1.5 and later, the reads are trimmed when a B is encountered at either end of the reads in the input file if the **Trim reads** option is checked.

Small samples of three kinds of files are shown below. The names of the reads have no influence on the quality score format:

NCBI/Sanger Phred scores:

```
@SRR001926.1 FC00002:7:1:111:750 length=36
TTTTTGTAAGGAGGGGGGTCATCAAAATTTGCAAAA
+SRR001926.1 FC00002:7:1:111:750 length=36
IIIIIIIIIIIIIIIIIIIIIIIIIFIIII'IB<IH
@SRR001926.7 FC00002:7:1:110:453 length=36
TTATATGGAGGCTTTAAGAGTCATAGGTTGTTCCCC
+SRR001926.7 FC00002:7:1:110:453 length=36
IIIIIIIIIII:'III?=IIIIII+&III/3I8F/&
```

Illumina Pipeline 1.2 and earlier (note the question mark at the end of line 4 - this is one of the values that are unique to the old Illumina pipeline format):

```
@SLXA-EAS1_89:1:1:672:654/1
GCTACGGAATAAAACCAGGAACAACAGACCCAGCA
+SLXA-EAS1_89:1:1:672:654/1
ccccccccccccccccccc]c``cVcZccbSYb?
@SLXA-EAS1_89:1:1:657:649/1
GCAGAAAATGGGAGTGAAAATCTCCGATGAGCAGC
+SLXA-EAS1_89:1:1:657:649/1
ccccccccccbccbccb``cccbcccZcc`^bR^`
```

The formulas used for converting the special Solexa-scale quality scores to Phred-scale:

$Q_{phred} = -10 \log_{10} p$

$Q_{solexa} = -10 \log_{10} \frac{p}{1-p}$

A sample of the quality scores of the Illumina Pipeline 1.3 and 1.4:

```
@HWI-E4_9_30WAF:1:1:8:178
GCCAGCGGCGCAAAATGNCGGCGGCGATGACCTTC
+HWI-E4_9_30WAF:1:1:8:178
babaaaa\ababaaaaREXabaaaaaaaaaaaaaa
@HWI-E4_9_30WAF:1:1:8:1689
GATGGAGATCTCGACCTNATAGGTGCCCTCATCGG
+HWI-E4_9_30WAF:1:1:8:1689
aab`]_aaaaaaaaa[ER`abaaa\aaaaaaaa[
```

Note that it is not possible to see from that data itself that it is actually not Illumina Pipeline 1.2 and earlier, since they use the same range of ASCII values.

To learn more about ASCII values, please see http://en.wikipedia.org/wiki/Ascii#ASCII_printable_characters.

6.3.2 PacBio

Choosing the PacBio import will open the dialog shown in figure 6.12.

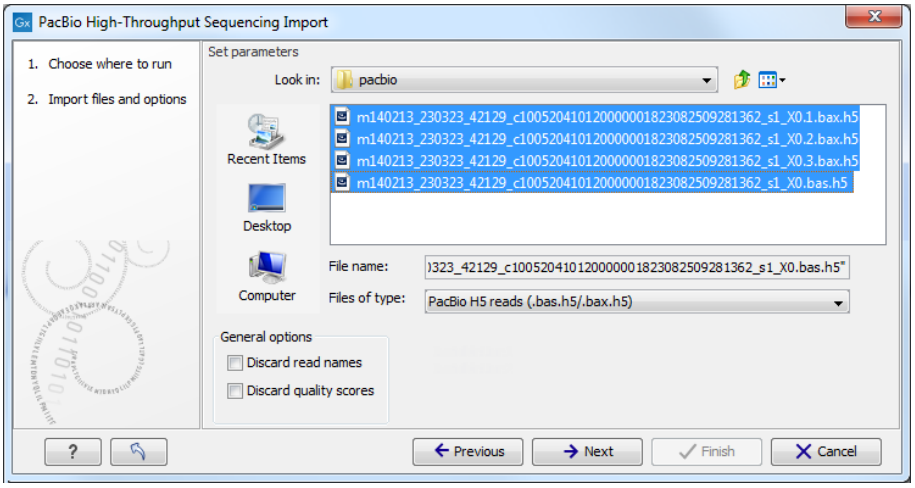


Figure 6.12: *Importing data from PacBio. ".bas.h5", ".fastq" and ".fasta" files are supported*

We support import of three file formats containing PacBio reads:

- H5 files (`.bas.h5/.bax.h5`) which contain one of two things. .bas.h5 files produced by instruments prior to PacBio RS II contain sequencing data such as reads and quality scores. .bas.h5 files from more recent PacBio instruments contain a list of .bax.h5 files where the actual sequencing data is stored. When importing H5 files, the user needs to select both the .bas.h5 file and all the accompanying .bax.h5 files belonging to a data set.
- Fastq files (`.fastq`) which contain sequence data and quality scores. Compressed Fastq (`.fastq.gz`) files are also supported.
- Fasta files (`.fasta`) which contain sequence data. Compressed Fasta (`.fasta.gz`) files are also supported.

Under **General options** you have the following choices:

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard read names to save disk space.
- **Discard quality scores**. Quality scores can be visualized in the mapping view and used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. Discarding quality scores will reduce both disk space usage and memory consumption. As PacBio quality scores currently contain very little information, we recommend that you discard them. When importing Fasta files, this option is not available, since Fasta files do not contain quality scores.

Click **Next** and choose how the result of the import should be handled. We recommend choosing **Save** which will save the results directly to the disk.

When opening the "Element info" of sequence lists imported with the PacBio importer, the item "Platform" will display the mention PACBIO. For PacBio reads imported without the PacBio importer, it is possible to edit that field to "PACBIO" by clicking **Edit** next to the "Read Group" section in the Element Info view. Having the platform set to PacBio will ensure that the read mapper will perform better on PacBio reads.

6.3.3 Fasta read files

The **Fasta** importer is designed for high volumes of read data such as high-throughput sequencing data (NGS reads). When using this import option the read names can be included but the descriptions from the fasta files are ignored.

For import of other fasta format data, such as reference sequences, please use the**Standard Import** ( ) as this import format also includes the descriptions. To have a reference in track format, use the **Tracks** ( ) option and set the "Type of file to import" to FASTA.

The dialog for importing data in fasta format is shown in figure 6.13.

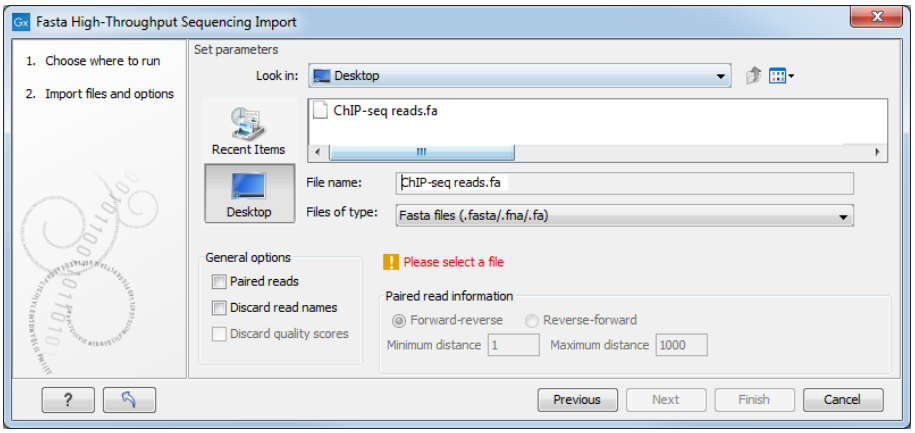


Figure 6.13: *Importing data in fasta format.*

Compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. For paired import, the Workbench expects the forward reads to be in one file and the reverse reads in another. The Workbench will sort the files before import and then assume that the first and second file belong together, and that the third and fourth file belong together etc. At the bottom of the dialog, you can choose whether the ordering of the files is **Forward-reverse** or **Reverse-forward**. As an example, you could have a data set with two files: `sample1_fwd` containing all the forward reads and `sample1_rev` containing all the reverse reads. In each file, the reads have to match each other, so that the first read in the `fwd` list should be paired with the first read in the `rev` list. Note that you can specify the insert sizes when importing paired read data. If you have data sets with different insert sizes, you should import each data set individually in order to be able to specify different insert sizes. Read more about handling paired data in section 6.3.7.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.
- **Discard quality scores**. This option is not relevant for fasta import, since quality scores are not supported.

Click **Next** to adjust how to handle the results (see section 8.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 8.3).

6.3.4 Sanger sequencing data

Although traditional sequencing data (with chromatogram traces like abi files) is usually imported using the standard **Import** (), see section 6, this option has also been included in the High-Throughput Sequencing Data import. It is designed to handle import of large amounts of sequences, and there are three differences from the standard import:

- All the sequences will be put in one sequence list (instead of single sequences).
- The chromatogram traces will be removed (quality scores remain). This is done to improve performance, since the trace data takes up a lot of disk space and significantly impacts speed and memory consumption for further analysis.
- Paired data is supported.

With the standard import, it is practically impossible to import up to thousands of trace files and use them in an assembly. With this special High-Throughput Sequencing import, there is no limit. The import formats supported are the same: ab, abi, ab1, scf and phd.

For all formats, compressed data in gzip format is also supported (.gz).

The dialog for importing data Sanger sequencing data is shown in figure 6.14.

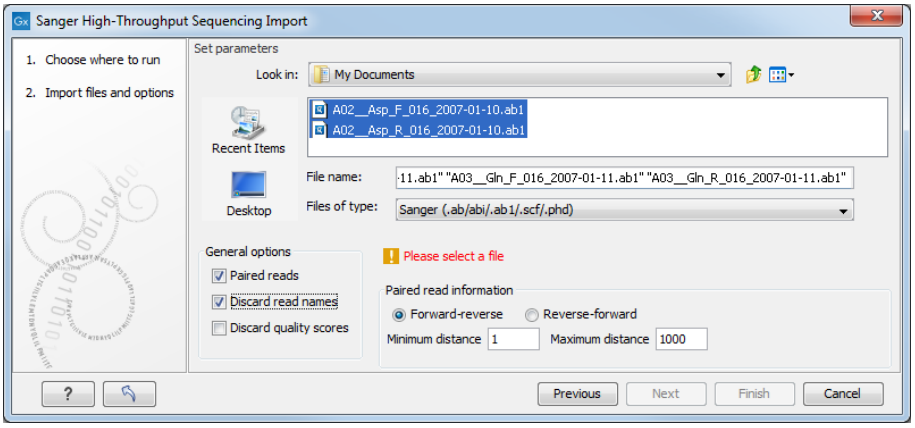


Figure 6.14: *Importing data from Sanger sequencing.*

The **General options** to the left are:

- **Paired reads**. The Workbench will sort the files before import and then assume that the first and second file belong together, and that the third and fourth file belong together etc. At the bottom of the dialog, you can choose whether the ordering of the files is **Forward-reverse** or **Reverse-forward**. As an example, you could have a data set with two files: `sample1_fwd` for the forward read and `sample1_rev` for the reverse reads. Note that you can specify the insert sizes when importing paired read data. If you have data sets with different insert sizes, you should import each data set individually in order to be able to specify different insert sizes. Read more about handling paired data in section 6.3.7.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption.

Click **Next** to adjust how to handle the results (see section 8.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 8.3).

6.3.5 Ion Torrent

Choosing the Ion Torrent import will open the dialog shown in figure 6.15.

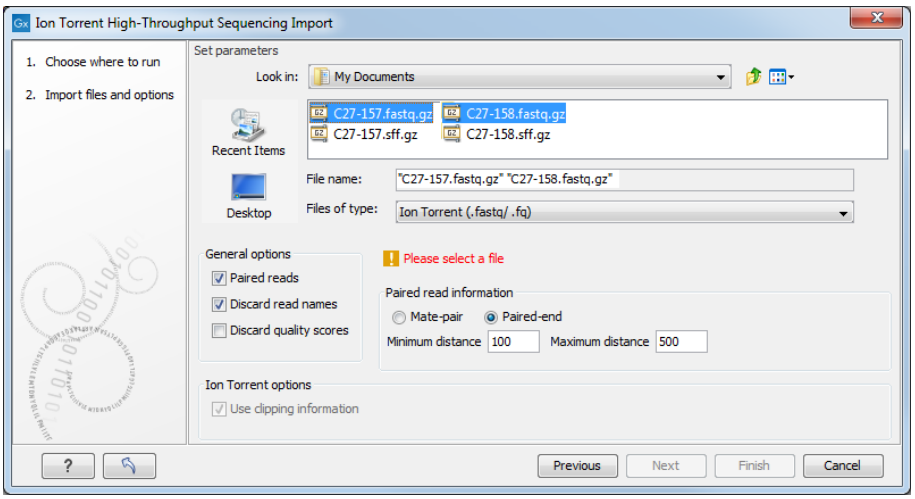


Figure 6.15: *Importing data from Ion Torrent.*

We support import of two kinds of data from the Ion Torrent system:

- SFF files (`.sff`) providing extra information on adapters or regions of low quality

- Fastq files (`.fastq`). Quality scores are expected to be in the NCBI/Sanger format (see section 6.3.1). Compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. The *Biomedical Genomics Workbench* supports both paired end and mate pair protocols.

  Paired end Paired end data from Ion Torrent comes in two files per data set. The first file in is assumed to contain the first reads of the pair, and the second file is assumed to contain the second read in a pair. On import, the orientation of the reads is set to forward - reverse. When the reads have been imported, there will be one file with intact pairs, and one file where one part of the pair is missing (in this case, "single" is appended to the file name). The Workbench connects the right sequences together in the pair based on the read name. Read more about handling paired data in section 6.3.7.

  Mate pair Mate pair reads from Ion torrent are Reverse-Forward and usually between 2,000 and 10,000 bp. The mate pair protocol for Ion Torrent entails that the two reads are separated by a linker sequence. During import of paired data, the linker sequence is removed and the two reads are separated and put into the same sequence list. You can change the linker sequence in the **Preferences** (in the **Edit** menu) under **Data**. When looking for the linker sequence, the Workbench requires 80 % of the maximum alignment score, using the following scoring scheme: matches = 1, mismatches = -2 and indels = -3. Some of the sequences may not have the linker in the middle of the sequence, and in that case the partial linker sequence is still removed, and the single read is put into a separate sequence list. Thus when you import Ion Torrent mate pair data, you may end up with two sequence lists: one for paired reads and one for single reads. Note that for de novo assembly projects, only the paired list should be used since the single reads list may contain reads where there is still a linker sequence present but only partially due to sequencing errors. Read more about handling paired data in section 6.3.7.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. If you have selected the fna/qual option and choose to discard quality scores, you do not need to select a .qual file.

For sff files, you can also decide whether to use the clipping information in the file or not.

6.3.6 Complete Genomics

With *Biomedical Genomics Workbench 5.0.1* you can import *evidence* and *variation* files from Complete Genomics.

The variation files can be imported as tracks (see section 6.2).

The evidence files can be imported using the SAM/BAM importer, see section 6.3.8.

In order to import the evidence data file it need to be converted first. This is achieved using the CGA tools that can be downloaded from http://www.completegenomics.com/sequence-data/cgatools/.

The procedure for converting the data is the following.

1. Download the human genome in fasta format and make sure the chromosomes are named `chr<number>.fa`, e.g. `chr9.fa`.
2. Run the **fasta2crr** tool with a command like this:
   `cgatools fasta2crr --input chr9.fa --output chr9.crr`
3. Run the **evidence2sam** tool with a command like this:
   `cgatools evidence2sam --beta -e evidenceDnbs-chr9-.tsv -o chr9.sam -s chr9.crr`
   where the .tsv file is the evidence file provided by Complete Genomics (you can find sample data sets on their ftp server: ftp://ftp.completegenomics.com/).
4. **Import** () the fasta file from 1. into the Workbench.
5. Use the SAM/BAM importer (section 6.3.8) to import the file created by the evidence2sam tool.

Please refer to the CGA documentation for a description about these tools.

6.3.7 General notes on handling paired data

During import, information about paired data (distances and orientation) can be specified (see figure 6.9) and stored by the *Biomedical Genomics Workbench*. All subsequent analyses automatically take differences in orientation into account. Once imported, both reads of a pair will be stored in the same sequence list. The forward and reverse reads (e.g. for paired-end data) simply alternate so that the first read is forward, the second read is the mate reverse read; the third is again forward and the fourth read is the mate reverse read and so on. When manipulating sequence lists with paired data, be careful not break this order.

You can view and edit the orientation of the reads after they have been imported by opening the read list in the Element information view (), see section 10.4 as shown in figure 6.16.

Figure 6.16: *The paired orientation and distance.*

In the **Paired status** part, you can specify whether the *Biomedical Genomics Workbench* should treat the data as paired data, what the orientation is and what the preferred distance is. The orientation and preferred distance is specified during import and can be changed in this view. If the "Paired sequences" box is unchecked, the sequences will be handled as single (non paired) data.

Note that the **paired distance** measure that is used throughout the *Biomedical Genomics Workbench* is always *including the full read sequence*. For paired-end libraries it means from the beginning of the forward read to the beginning of the reverse read.

6.3.8 SAM and BAM mapping files

The *Biomedical Genomics Workbench* supports import and export of files in SAM (Sequence Alignment/Map) and BAM format, which are designed for storing large nucleotide sequence alignments. Read more and see the format specification at http://samtools.sourceforge.net/

The *Biomedical Genomics Workbench* includes support for importing SAM and BAM files from **Complete Genomics**.

Note! If you wish to import the reads in a SAM/BAM file as a sequence list, disregarding any mapping information, please use the Standard import tool instead (see section 6.1).

For a detailed explanation of the SAM and BAM files exported from *Biomedical Genomics Workbench*, please see Appendix I.

Input data for importing a mapping from a SAM/BAM file

To import a mapping from a SAM/BAM file containing mapping data into the Workbench, you need to:

- Provide the SAM/BAM file
- Specify the reference sequences that are referred to within that file. The references can either be sequences already imported into the Workbench, or, if appropriately recorded in the SAM/BAM file, can be fetched from URLs specified in the SAM/BAM file.

The mapping is built up within the Workbench using the reference sequence data, the reads and the information from the SAM/BAM file about how the reads are associated with a particular reference.

Data created in the Workbench after importing a SAM/BAM mapping file

- Reads recorded as mapping to a particular reference that is known inside the Workbench are imported as part of the mapping for that reference.
- Reads recorded as not mapping to any reference are imported into a sequence list.
  - If they are part of an intact pair, they are imported into a sequence list of paired data.
  - If they are single reads or a member of a pair that did not map while its mate did, they are imported into a sequence list containing single reads.

  One list is made per read group, with the potential that several such lists could be produced from a single mapping import. The sequence lists are given names of this form for single reads "<read group id> [read group sample] (single) un-mapped reads" and this form for paired reads "<read group id> [read group sample] (paired) un-mapped reads".

  If you do not wish to import the unmapped reads, deselect the **Import unmapped reads** option in the final step of the tool dialog.

- Reads recorded as mapping to a reference sequence that is **not** known within the Workbench are not imported.

When setting up the import, you are given the option of creating a track-based mapping, or a stand-alone mapping. In the latter case, if there is only one reference sequence, the result will be a single read mapping ( ). When there is more than one reference sequence, a multi- mapping object ( ) is created.

Please note that mappings within the *Biomedical Genomics Workbench* do not allow for an individual read sequence to map to more than one location. In cases where a SAM/BAM file contains multiple alignment records for a single read, only one such record will be used to build the mapping.

Running the SAM/BAM Mapping Files importer

Click on the Import button on the toolbar or go to:

File | Import ( ) | SAM/BAM Mapping Files ( )

This will open a dialog where you select the SAM/BAM file to import as well as the reference sequences to be used (Figure 6.17).

When you select the reference sequence(s) two options exist:

1. Select a matching reference sequence that has already been imported into the Workbench. Click on the "Find in folder" icon ( ) to localize the reference sequence.
2. If the SAM/BAM file already contains information about where to find the reference sequence, tick the "Download references" box to automatically download the reference sequence.

The selected reference sequence(s) will be listed under "References in files" with "Name", "Length", and "Status". Whenever the correct reference sequence (with the correct name and sequence length) has been selected the "Status" field will indicate this with an "OK". The length of your reference sequence must **match exactly** the length of the reference specified in the SAM/BAM file. The name is more flexible as it allows a range of different "synonyms" (with no distinction between capital and lowercase letters). E.g. for chromosome 1 the allowed synonyms would be: 1, chr1, chromosome_1, nc_000001, for chromosome M: m, mt, chrm, chrmt, chromosome_m, chromosome_mt, nc_001807, for chromosome X: x, chrx, chromosome_x, nc_000023, and for chr Y: y, chry, chromosome_y, nc_000024.

If there are inconsistencies in the names or lengths of the reference sequences being chosen and those recorded in the SAM/BAM file, an entry will appear in the "Status" column indicating this (for example, "Length differs" or "Input missing")[1].

Unmatched reads (reads that are mapped to an unmatched reference e.g. a SAM reference for which there is no CLC reference counterpart) are not imported. The same is the case whenever inconsistencies have occurred with respect to name or length. The log lists all mapping data or

[1] If you are using a CLC Genomics Server to import files located on the Server (rather than locally), then checks for corresponding reference names and lengths cannot be carried out, so nothing will be reported in this section of the Wizard. This means you will be able to continue to launch the import with correct or incorrect reference sets specified. However, any inconsistencies in these will lead to the import task failing with an error related to this.

unmatched reads that were not imported and marks whether import failed because of unmatched reads being present in the SAM/BAM file or because of inconsistencies in name/length.

Some notes regarding reference sequence naming Reference sequences in a SAM/BAM file **cannot contain spaces**. If the name of a reference sequence in the Workbench contains spaces, the Workbench assume that the names of the references in the SAM file will be the same as the names of the References within the Workbench, but with all spaces removed. For exapmple, if your reference sequence in the Workbench was called `my reference sequence`, the Workbench would recognize a reference in the SAM file as the appropriate reference if it was of the same length and had the name `myreferencesequence`.

Neither the `@` character nor the `=` character are allowed within reference sequence names in SAM files. Any instances of these characters in the name of a reference sequence in the Workbench will be replaced with a _ for the sake of identifying the appropriate reference when importing a SAM or BAM file. For example, if a reference sequence in the Workbench was called `my=reference@sequence`, the Workbench would recognize a reference in the SAM file as the appropriate reference if it was of the same length and had the name `my_reference_sequence`.

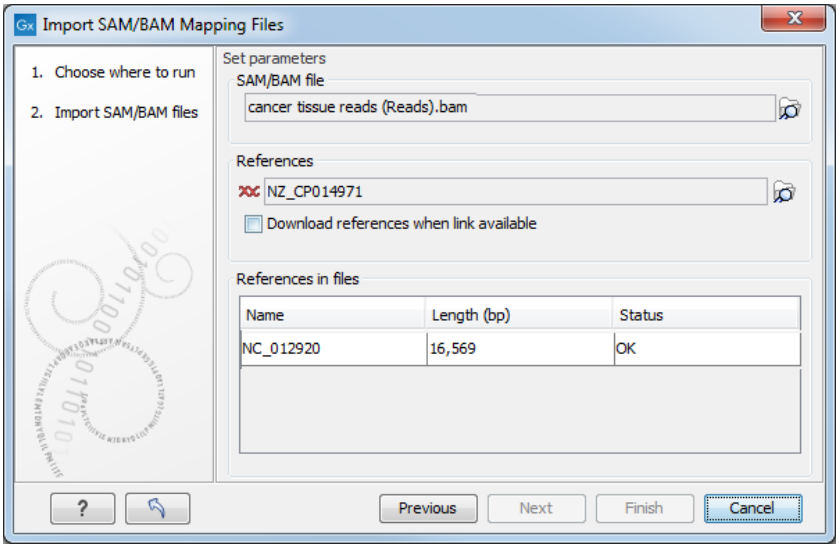


Figure 6.17: *Defining SAM/BAM file and reference sequence(s).*

Click **Next** to specify how to handle the results (Figure 6.18). Under **Output options** the "Save downloaded reference sequence" will be enabled if the "Download references" box was ticked in the previous step (which would be the case when the SAM/BAM file contained information about where to find the reference sequence e.g. if the SAM/BAM file came from an external provider).

Ticking the "Create Reads Track" box results in the generation of a track-based mapping. Alternatively, the "Create Stand-Alone Read Mapping" results in a normal read mapping file. By ticking the "Import unmapped reads" box, a sequence list of the unmapped reads will be created. To avoid importing unmapped reads, untick this box.

We recommend choosing **Save** in order to save the results directly to a folder, as you will probably wish to save the data anyway before proceeding with your analysis. For further information about how to handle the results, (see section 8.2).

Note that this import operation is very memory-consuming for large data sets, and particularly those with many reads marked as members of broken pairs in the mapping.

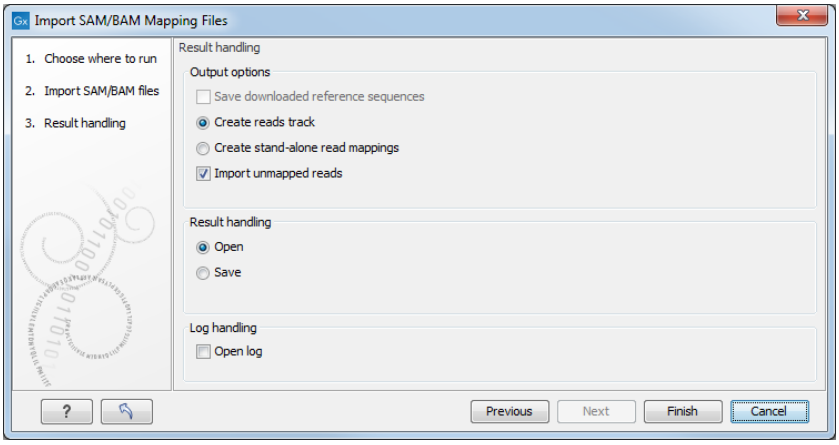


Figure 6.18: *Specify the result handling.*

6.4 Import RNA spike-in controls

The *Biomedical Genomics Workbench* has a dedicated tool for importing RNA spike-in control data: **Import | RNA Spike-ins**

The wizard offers the option to import a standard ERCC file as provided by Thermo Fisher Scientific, or a custom made one (figure 6.19).

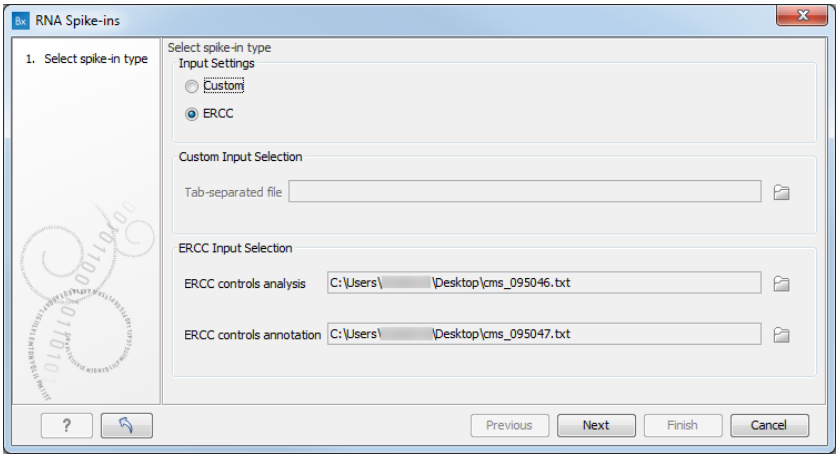


Figure 6.19: *The RNA spike-in controls importer.*

To import a standard ERCC file, look for **ERCC Controls Analysis** and **ERCC Control Annotation** files on the Thermo Fisher Scientific website, download both *.txt files on your computer, and start the importer. Select the option "ERCC" and specify the location of the analysis and annotation files in the relevant "ERCC Input Selection" fields at the bottom of the wizard.

For **custom-made spike-in controls**, choose the "Custom" option and specify in the "Custom Input Selection" field a tab-separated file (*.tsv or *.txt) containing the spike-in data organized as such: sequence name in the first column, nucleotide sequence in the second column, followed by as many columns as necessary to contain the concentrations of the spike-in measures in attomoles/microliters. Concentrations must not contain commas: write 15000 instead of 15,000. Remove any white space and save the table as a tab-separated TSV or TXT file on your computer.

It is also possible to import Lexogen Spike-in RNA Variant Control Mixes by modifying the SIRV files to fit the custom file requirements. Download the **SIRV sequence design overview (XLSX)**

from the Lexogen website and open it in Excel. In the annotation column, "c" designate the data that should be imported ("i" is under-annotated while "0" is over-annotated). Filter the table to only keep the rows having a 1 in the "c" column, then keep only - and in that order - the sequence name, nucleotide sequence and concentration columns of the remaining rows. Reformat the values to numerical values in attomoles/microliters before saving the table as a *.tsv file. Import the file in the workbench using the "Custom" option.

Once a spike-in file is specified, click **Next** and choose to **Save** the file in the Navigation Area for later use in RNA-Seq Analysis.

6.5 Import Primer Pairs

The **Import Primer Pairs** importer can import descriptions of primer locations from a generic text format file or from a QIAGEN gene panel primer file.

The **Import Primer Pairs** can be found in the toolbar:

Import () | **Import Primer Pairs** ()

This will open the wizard shown in figure 6.20. The first step is to select the data to import.

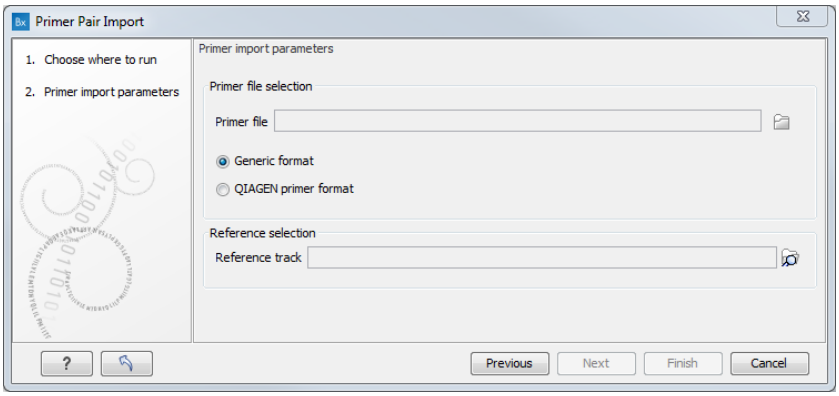


Figure 6.20: *Select files to import.*

- **Primer File** Click on the folder icon in the right side to select your primer pair location file.

  There are two primer pair formats that can be imported by the Workbench.

  - **Generic Format** Select this option for primer location files with the exception of QIAGEN gene panel primers. Provide your primer location information in a tab delimited text file with the following columns:

    * Column 1: reference name
    * Column 2: primer1 first position (5'end) on reference
    * Column 3: primer1 last position (3'end) on reference
    * Column 4: primer2 first position (5'end) on reference
    * Column 5: primer2 last position (3'end) on reference
    * Column 6: amplicon name

    Note: Primer position intervals are left-open and right-closed, so the leftmost position of the primer on the reference (column 2 and 5) should have one subtracted.

An example of the format expected for each row is:

```
chr1    42    65    142    106    Amplicon1
```

Indicating forward and reverse primers covering the reference nucleotides [43, 65] and [107, 142].

- **QIAGEN Primer Format** Use this option for importing information about QIAGEN gene panel primers.

- **Reference Track** Use folder icon in the right side to select the relevant reference track.

Click on the button labeled **Next** to go to the wizard step choose to save the imported primer location file.

6.6 Data export

The exporter can be used to:

- Export bioinformatic data in most of the formats that can be imported. There are a few exceptions (see section H.1).
- Export one or more data elements at a time to a given format. When multiple data elements are selected, each is written out to an individual file, unless compression is turned on, or "Output as single file" is selected.

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

File | **Export (** **)**

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

File | **Export with Dependent Elements**

This tool is described further in section 6.6.2.

The general steps when configuring a standard export job are:

- (Optional) Select the data to export in the **Navigation Area**.
- Start up the exporter tool via the Export button in the toolbar or using the **Export** option under the File menu.
- Select the format the data should be exported to.
- Select the data to export, or confirm the data to export if it was already selected via the **Navigation Area**.
- Configure the parameters. This includes compression, multiple or single outputs, and naming of the output files, along with other format-specific settings where relevant.
- Select where the data should be exported to.

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

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

Selecting a format to export to. When data is pre-selected in the **Navigation Area** before launching the export tool you will see a column in the export interface called **Supported formats**. Formats that the selected data elements can be exported to are indicated by a "Yes" in this column. Supported formats will appear at the top of the list of formats (figure 6.21).

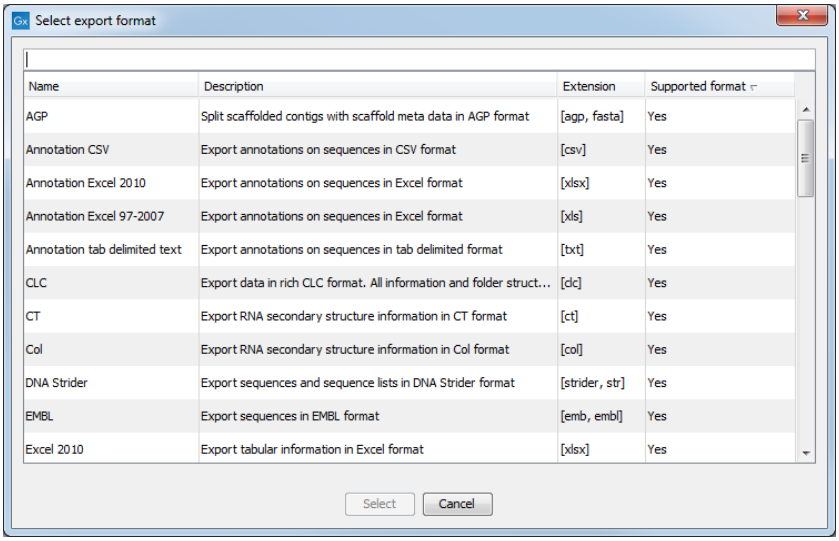


Figure 6.21: *The Select exporter dialog where sequence lists were pre-selected in the Navigation Area before launching the export tool. Here, the formats sequence lists can be exported to are listed at the top, with a Yes in the Selected formats column. Other formats are found below, with No in this column.*

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

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

Only one export format is available if you select a folder to be exported. This is described in more detail in section 6.6.1.

Finding a particular format in the list. You can quickly find a particular format by using the text box at the top of the exporter window as shown in figure 6.22, where formats that include the term VCF are searched for. This search term will remain in place the next time the Export tool is launched. Just delete the text from the search box if you no longer wish only the formats with that term to be listed.

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

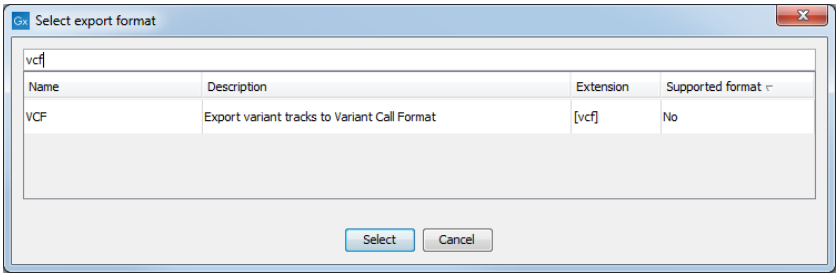


Figure 6.22: *The text field has been used to search for VCF format in the Select exporter dialog.*

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

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

In figure 6.23 we show the selection of a variant track for export to VCF format.

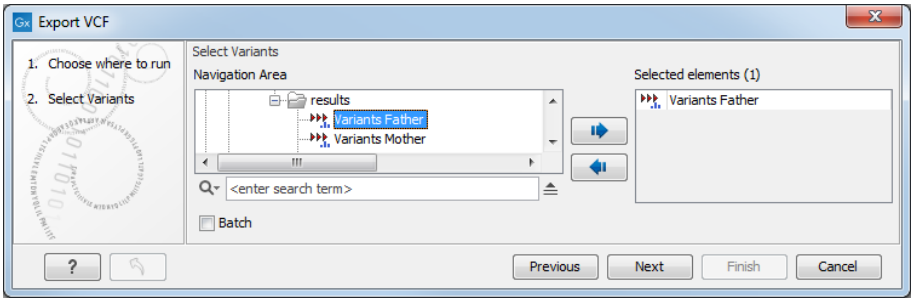


Figure 6.23: *The Select exporter dialog. Select the data element(s) to export.*

The parameters under **Basic export parameters** and **File name** are offered when exporting to any format.

There may be additional parameters for particular export formats. This is illustrated here with the VCF exporter, where a reference sequence track must be selected (see figure 6.24). To learn more about VCF specific export format, please read section H.1.7.

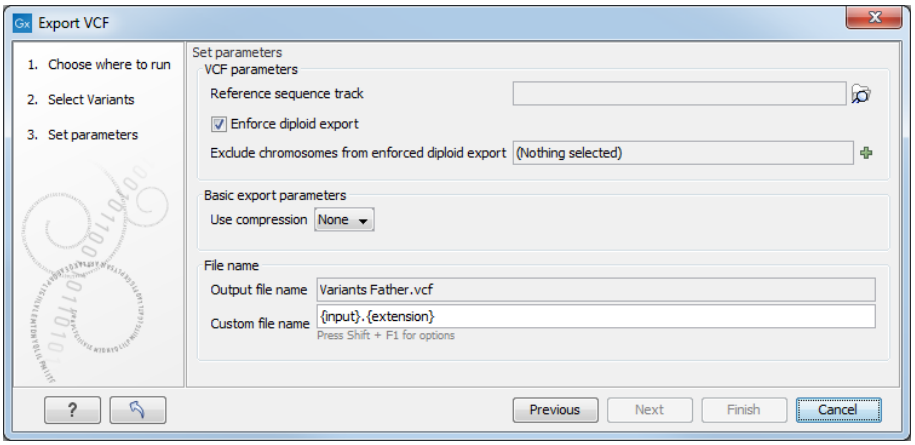


Figure 6.24: *Set the export parameters. When exporting in VCF format, a reference sequence track must be selected.*

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.

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.

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.

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

When exporting just one data element, or exporting to a zip file, the desired filename could just be typed in the Custom file name box.

When working with the export of multiple files, using some combination of the terms shown by default in this field and in figure 6.27 are recommended. Clicking in the **Custome file name** field with the mouse and then simultaneously pressing the Shift + F1 keys bring up a list of the available terms that can be included in this field.

The following placeholders are available:

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

We will look at an example to illustrate this: In this example we would like to change the export file format to .fasta in a situation where .fa was the default format that would be used if you kept the default file extension suggestion ("{2}"). To do this replace "{2}" with ".fasta" in the "Custom file name field". You can see that when changing "{2}" to ".fasta" , the file name extension in the "Output file name" field automatically changes to the new format (see figure 6.25).

When deciding on an output name, you can choose any combination of the different placeholders as well as custom names and punctuation, as in `{input}({day}-{month}-{year})`. Another example of a meaningful name to a variant track could be `{2} variant track` as shown in figure 6.26. If your workflow input is named `Sample 1`, the result would be "Sample 1 variant track".

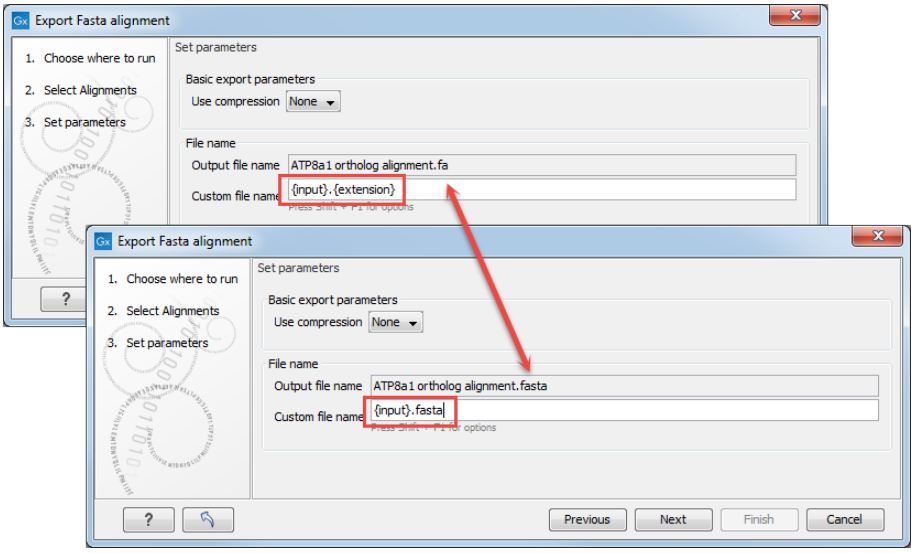


Figure 6.25: *The file name extension can be changed by typing in the preferred file name format.*

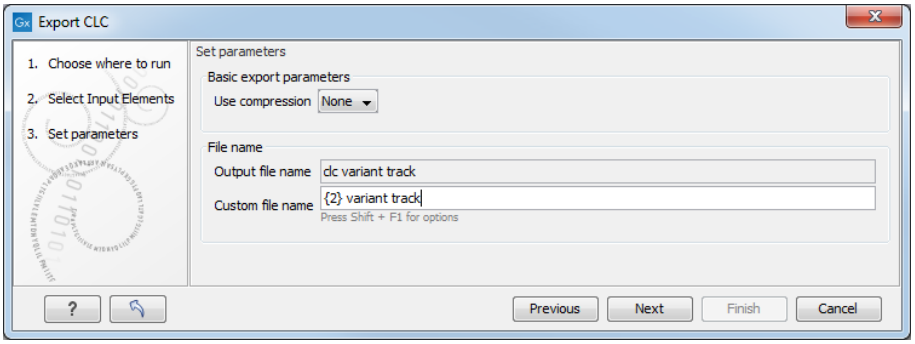


Figure 6.26: *Providing a custom name for the result.*

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

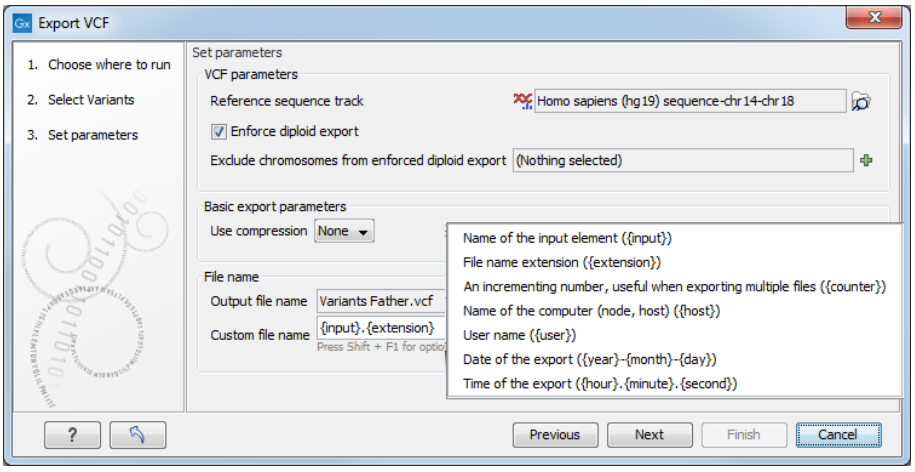


Figure 6.27: *Use the custom file name pattern text field to make custom names.*

The last step is to specify the exported data should be saved.

A note about decimals and Locale settings. When exporting to CSV and tab delimited files, decimal numbers are formatted according to the Locale setting of the Workbench (see section 4.1). If you open the CSV or tab delimited file with spreadsheet software like Excel, you should make sure that both the Workbench and the spreadsheet software are using the same Locale.

6.6.1 Export of folders and multiple elements in CLC format

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

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

Note! When exporting multiple files, the names will be listed in the "Output file name" text field with only the first file name being visible and the rest being substituted by "...", but will appear in a tool tip if you hover the mouse over that field (figure 6.28).

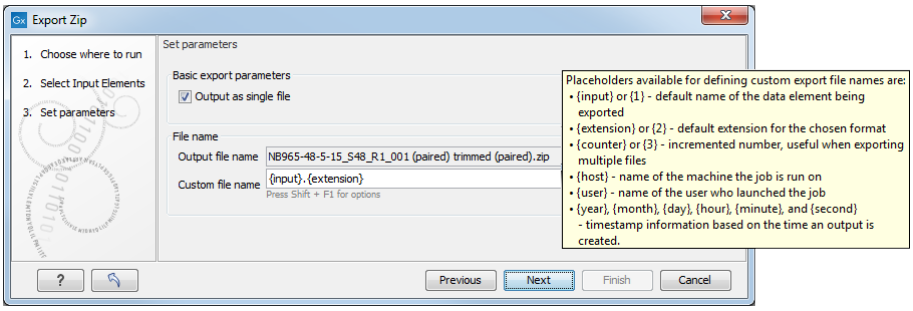


Figure 6.28: *The output file names are listed in the "Output file name" text field.*

6.6.2 Export of dependent elements

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

To export a data element with its dependent elements:

- Select the parent data element (like an alignment) in the **Navigation Area**.
- Start up the exporter tool by going to **File** | **Export with Dependent Elements**.
- Edit the output name if desired and select where the resulting zip format file should be exported to.

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

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

File | **Import** | **Standard Import**

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

6.6.3 Export history

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

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

- (Optional, but preferred) Select the data element (like an alignment) in the **Navigation Area**.
- Start up the exporter tool via the Export button in the toolbar or using the **Export** option under the File menu.
- Select the **History PDF** or History CSV as the format to export to (figure 6.29).
- Select the data to export, or confirm the data to export if it was already selected via the **Navigation Area**.
- Edit any parameters of interest, such as the Page Setup details, the output filename(s) and whether or not compression should be applied (figure 6.30).
- Select where the data should be exported to.
- Click on the button labeled **Finish**.

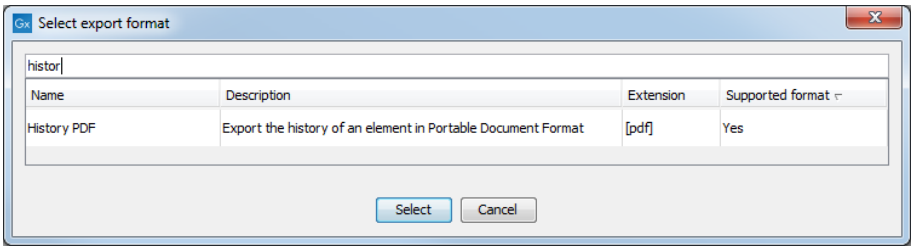


Figure 6.29: *Select "History PDF" for exporting the history of an element as a PDF file.*

6.6.4 The CLC format

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

A given data element in the Workbench can contain different types of data. This is reflected when exporting data, as the choice of different export formats can lead to the extraction of some parts of that data object rather than others. The part of the data exported reflects the type of data a given format can support. As a simple example, if you export the results of an alignment

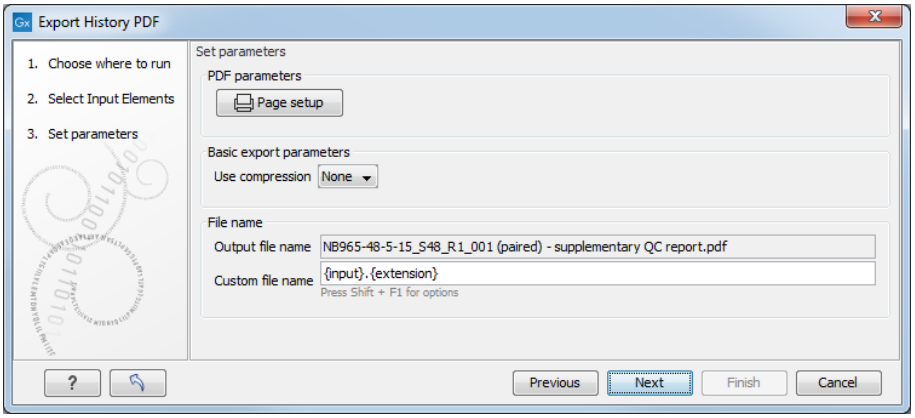


Figure 6.30: *When exporting the history in PDF, it is possible to adjust the page setup.*

to Annotation CSV format, you will get just the annotation information. If you exported to Fasta alignment format, you would get the aligned sequences in fasta format, but no annotations.

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

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

6.6.5 Backing up data from the CLC Workbench

Regular backups of your data are advisable.

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

Option 1: Backing up each CLC Data Location

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

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

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

Information about your data locations can also be found in an xml file called model_settings_300.xml This file is located in the settings folder in the user home area. Further details about this file and how it pertains to data locations in the Workbench can be found in the Deployment Manual: http://resources.qiagenbioinformatics.com/manuals/workbenchdeployment/current/index.php?manual=Changing_default_location.html.

Option 2: Export a folder of data or individual data elements to a CLC zip file

This option is for backing up smaller amounts of data, for example, certain results files or a whole data location, where that location contains smaller amounts of data. For data that takes up many gigabases of space, this method can be used, but it can be very demanding on space, as well as time.

Select the data items, including any folders, in the Navigation area of your Workbench and choose to export by going to:

File | **Export (**  **)**

and choosing ZIP format.

The zip file created will contain all the data you selected. You can later re-import the zip file into the Workbench by going to:

File | **Import (**  **)**

The only data files associated with the *Biomedical Genomics Workbench* not within a specified data location are BLAST databases. It is unusual to back up BLAST databases as they are usually updated relatively frequently and in many cases can be easily re-created from the original files or re-downloaded from public resources. If you do wish to backup your BLAST database files, they can be found in the folders specified in the BLAST Database Manager, which is started by going to: **Toolbox | BLAST | Manage BLAST databases** .

6.6.6 Export of tables

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

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

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

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

The Excel exporters, the CSV and tab delimited exporters, and the HTML exporter have been extended with the ability to export only a sub-set of columns from the object being exported. Uncheck the option "Export all columns" and click next to see a new dialog window in which columns to be exported can be selected (figure 6.31). You can choose them one by one or choose a predefined subset:

- All: will select all possible columns.
- None: will clear all preselected column.
- Default: will select the columns preselected by default by the software.

- Last export: will select all windows that were selected during the last export.
- Active editor (only if an active editor is currently open): the columns selected are the same than in the active editor window.

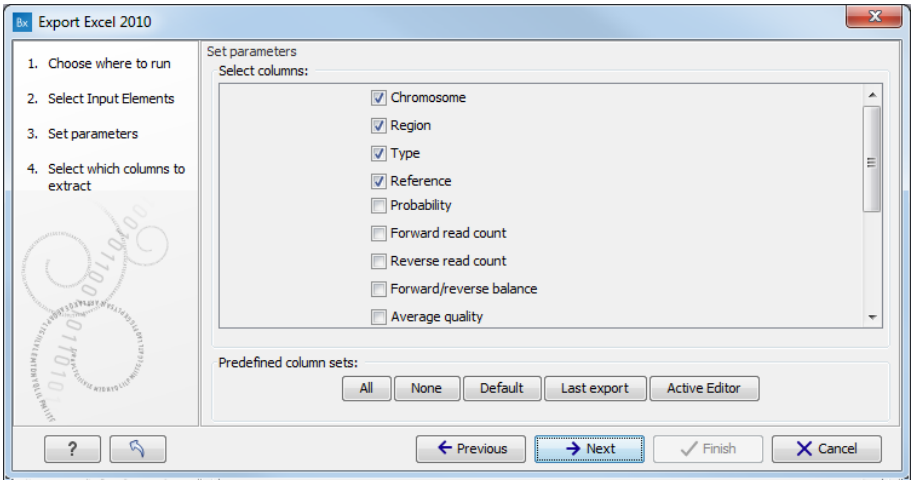


Figure 6.31: *Selecting columns to be exported.*

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

6.7 Export graphics to files

Biomedical Genomics Workbench supports export of graphics into a number of formats. This way, the visible output of your work can easily be saved and used in presentations, reports etc. The **Export Graphics** function ( ) is found in the **Toolbar**.

Biomedical Genomics Workbench uses a WYSIWYG principle for graphics export: What You See Is What You Get. This means that you should use the options in the Side Panel to change how your data, e.g. a sequence, looks in the program. When you export it, the graphics file will look exactly the same way.

It is not possible to export graphics of elements directly from the **Navigation Area**. They must first be opened in a view in order to be exported. To export graphics of the contents of a view:

select tab of View | Graphics ( ) on Toolbar

This will display the dialog shown in figure 6.32.

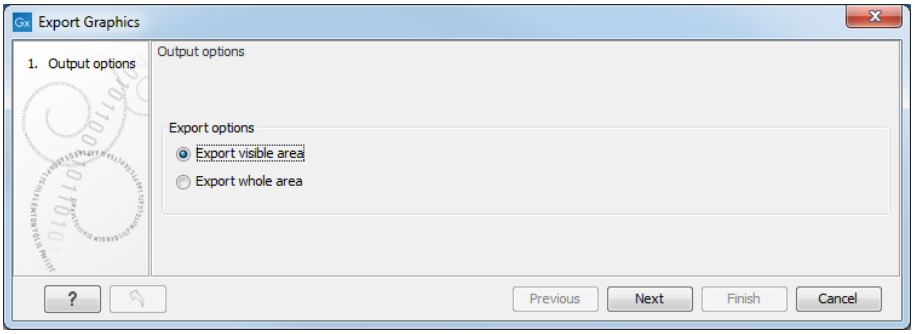


Figure 6.32: *Selecting to export whole view or to export only the visible area.*

In the following dialog, you can choose to:

- **Export visible area**, or
- **Export whole view**

These options are available for all views that can be zoomed in and out. In figure 6.33 is a view of a circular sequence which is zoomed in so that you can only see a part of it.

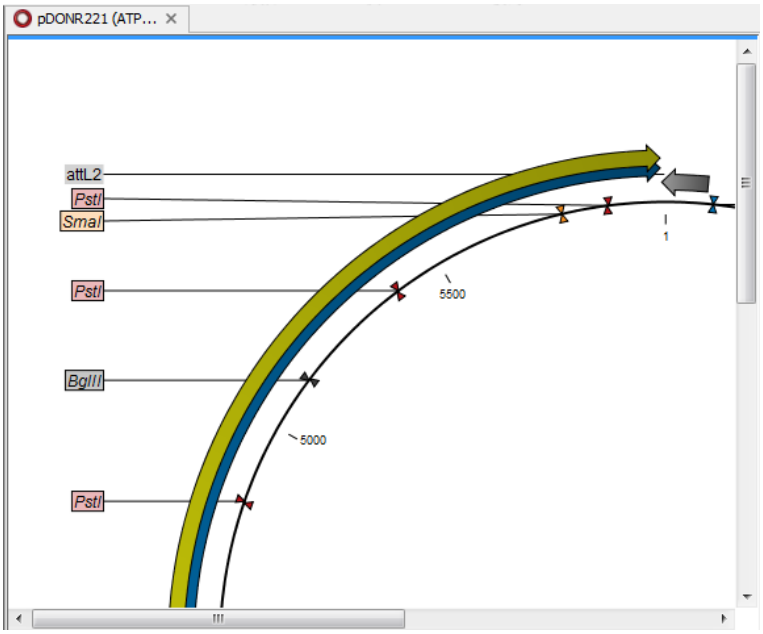


Figure 6.33: *A circular sequence as it looks on the screen when zoomed in.*

When selecting **Export visible area**, the exported file will only contain the part of the sequence that is *visible* in the view. The result from exporting the view from figure 6.33 and choosing **Export visible area** can be seen in figure 6.34.

On the other hand, if you select **Export whole view**, you will get a result that looks like figure 6.35. This means that the graphics file will also include the part of the sequence which is not visible when you have zoomed in.

For 3D structures, this first step is omitted and you will always export what is shown in the view (equivalent to selecting **Export visible area**).

Finally, choose a name and save location for the graphics file. Then you can either click **Next** or **Finish**, depending on what is available: clicking **Next** allows you to set further parameters for the graphics export, whereas clicking **Finish** will export using the parameters that you have set last time you made a graphics export in that file format (if it is the first time, it will use default parameters).

6.7.1 File formats

Biomedical Genomics Workbench supports the following file formats for graphics export:

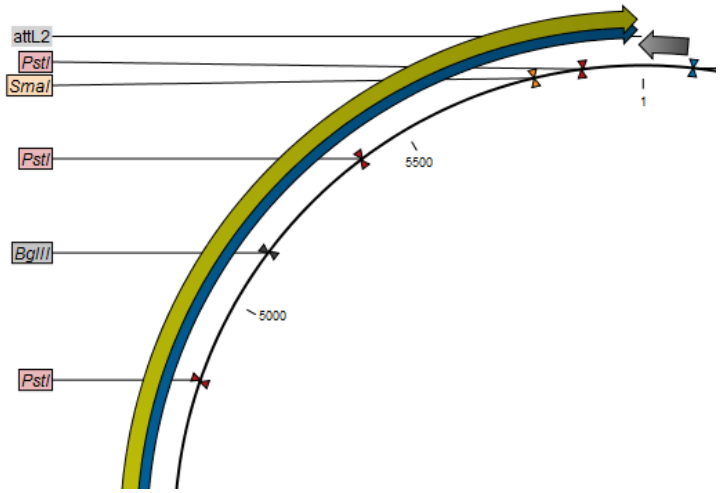


Figure 6.34: *The exported graphics file when selecting Export visible area.*

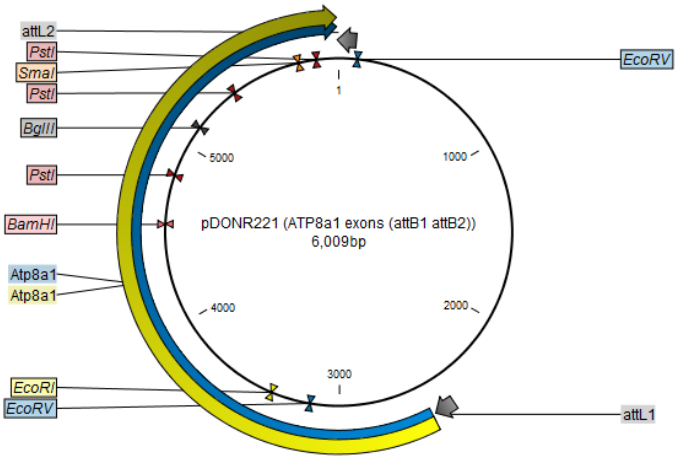


Figure 6.35: *The exported graphics file when selecting Export whole view. The whole sequence is shown, even though the view is zoomed in on a part of the sequence.*

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

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

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

Parameters for bitmap formats For bitmap files, clicking **Next** will display the dialog shown in figure 6.36.

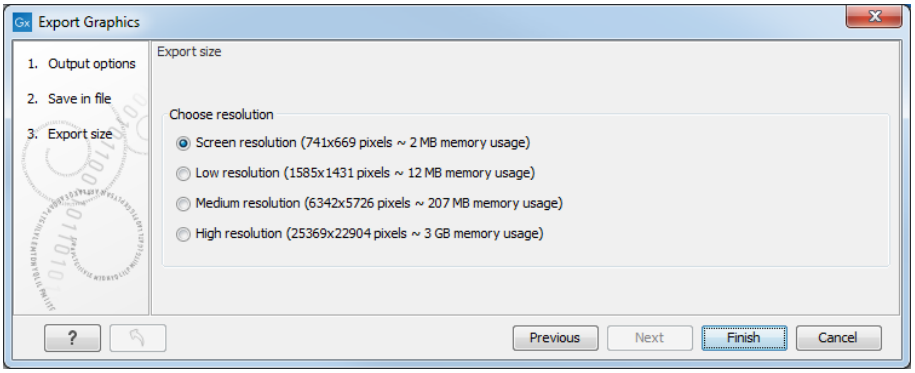


Figure 6.36: *Parameters for bitmap formats: size of the graphics file.*

You can adjust the size (the resolution) of the file to four standard sizes:

- Screen resolution
- Low resolution
- Medium resolution
- High resolution

The actual size in pixels is displayed in parentheses. An estimate of the memory usage for exporting the file is also shown. If the image is to be used on computer screens only, a low resolution is sufficient. If the image is going to be used on printed material, a higher resolution is necessary to produce a good result.

Vector graphics Vector graphic is a collection of shapes. Thus what is stored is information about where a line starts and ends, and the color of the line and its width. This enables a given viewer to decide how to draw the line, no matter what the zoom factor is, thereby always giving a correct image. This format is good for graphs and reports, but less usable for dot plots. If the image is to be resized or edited, vector graphics are by far the best format to store graphics. If you open a vector graphics file in an application such as Adobe Illustrator, you will be able to manipulate the image in great detail.

Graphics files can also be imported into the **Navigation Area**. However, no kinds of graphics files can be displayed in *Biomedical Genomics Workbench*. See section 6.1.1 for more about importing external files into *Biomedical Genomics Workbench*.

Parameters for vector formats For PDF format, the dialog shown in figure 6.37 will sometimes appear after you have clicked finished (for example when the graphics use more than one page, or there is more than one PDF to export).

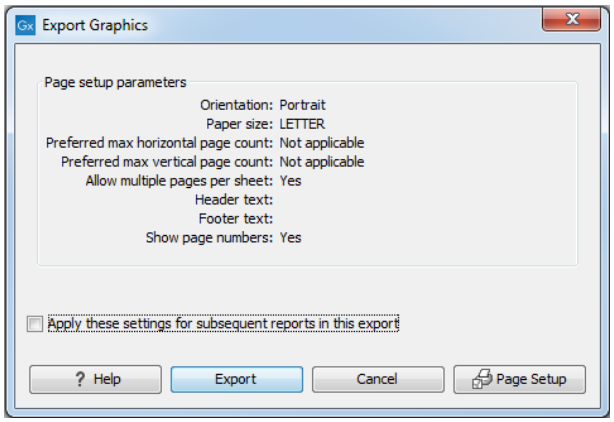


Figure 6.37: *Page setup parameters for vector formats.*

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

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

The page setup is only available if you have selected to export the whole view - if you have chosen to export the visible area only, the graphics file will be on one page with no headers or footers.

Exporting protein reports It is possible to export a protein report using the normal **Export** function ( ) which will generate a pdf file with a table of contents:

Click the report in the Navigation Area | Export ( ) in the Toolbar | select pdf

You can also choose to export a protein report using the **Export graphics** function ( ), but in this way you will not get the table of contents.

6.8 Export graph data points to a file

Data points for graphs displayed along the sequence or along an alignment or mapping can be exported to a semicolon-separated text file (csv format). An example of such a graph is shown in figure 6.38. This graph shows the coverage of reads in a read mapping.

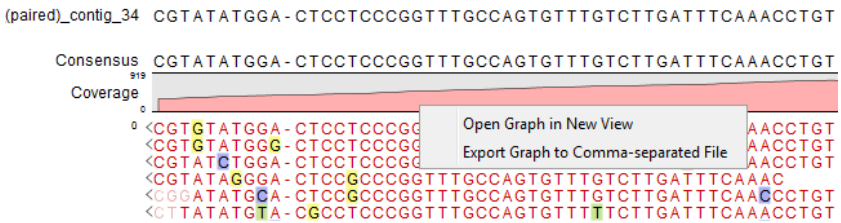


Figure 6.38: *A graph displayed along mapped reads. Right-click the graph to export the data points to a file.*

To export the data points for the graph, right-click the graph and choose **Export Graph to**

Comma-separated File. Depending on what kind of graph you have selected, different options will be shown: If the graph is covering a set of aligned sequences with a main sequence, such as read mappings and BLAST results, the dialog shown in figure 6.39 will be displayed. These kinds of graphs are located under **Alignment info** in the Side Panel. In all other cases, a normal file dialog will be shown letting you specify name and location for the file.

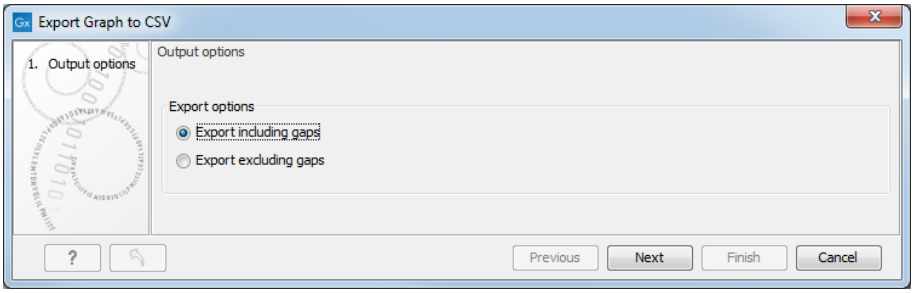


Figure 6.39: *Choosing to include data points with gaps*

In this dialog, select whether you wish to include positions where the main sequence (the reference sequence for read mappings and the query sequence for BLAST results) has gaps. If you are exporting e.g. coverage information from a read mapping, you would probably want to exclude gaps, if you want the positions in the exported file to match the reference (i.e. chromosome) coordinates. If you export including gaps, the data points in the file no longer corresponds to the reference coordinates, because each gap will shift the coordinates.

Clicking **Next** will present a file dialog letting you specify name and location for the file.

The output format of the file is like this:

```
"Position";"Value";
"1";"13";
"2";"16";
"3";"23";
"4";"17";
...
```

6.9 Copy/paste view output

The content of tables (reports, folder lists, and sequence lists) can be copy/pasted into different programs, where it can be edited. *Biomedical Genomics Workbench* pastes the data in tabulator separated format in various programs in which the copy/paste can be applied. For simplicity, we include one example of the copy/paste function from a **Folder Content** view to Microsoft Excel.

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

See figure 6.40.

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

The outcome might appear unorganized, but with a few operations the structure of the view in *Biomedical Genomics Workbench* can be produced. (Except the icons which are replaced by file references in Excel.)

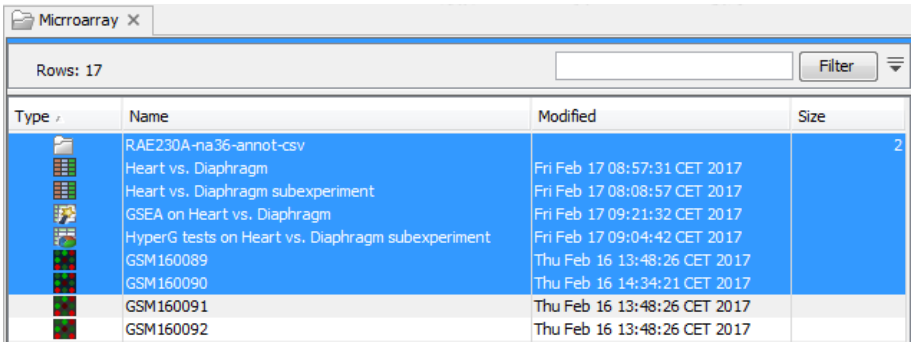


Figure 6.40: *Selected elements in a Folder Content view.*

Note that all tables can also be **Exported** ( ) directly in Excel format.

Chapter 7

Data download

Contents

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

7.1 SRA search

This section describes searches in SRA and the handling of search results. SRA is an NCBI maintained database of NGS data.

The SRA search view (figure 7.1) is opened in this way:

Download | Search for Reads in SRA ( **)**

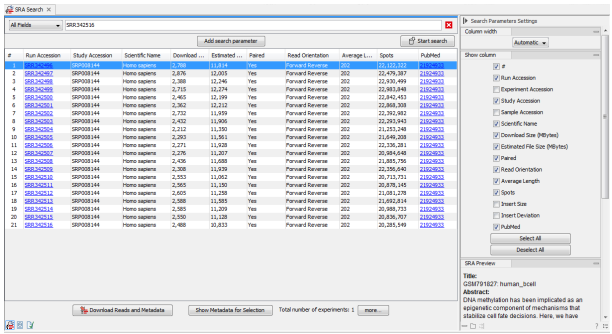


Figure 7.1: *The SRA search view.*

The tool queries the database with SRA accession number (see in figure 7.1) or various entries such as "hindgut" or "genometrakr". It is also possible to look for entries with certain properties, and to form more refined queries such as "paired-end RNA-Seq data from an Illumina HiSeq2500".

7.1.1 SRA search options

Search parameters Different types of search are available from a drop-down menu on the left hand side of the search bar.

Some special types are:

- **Modification/Publication date** This allows the narrowing of results to a particular range of dates specified as [MM] [YYYY]. For example 08 2016 to 08 2016 returns results published from the first day in August 2016 to the last day in August 2016.
- **Strategy** Provides an additional drop-down list of types of experiments e.g., RNA-Seq, ChIP-Seq, etc.
- **Library Selection** Provides an additional drop-down list of known library preparation methods, e.g. Poly(A) and Size fractionation.
- **Platform** Provides an additional drop-down list of NGS sequencing platforms e.g., Illumina, Ion Torrent. Note that download of data from some platforms such as Complete Genomics is not supported.
- **Instrument** Provides an additional drop-down list of individual NGS sequencing machines e.g., HiSeq X Ten, Ion Torrent PGM.
- **Paired Status** Choose between paired end and single end runs.
- **Availability** Choose between dbGaP or public. dbGaP refers to confidential data that can be searched through the tool and accessed upon request at NCBI.
- **PubMed** Choose between "has abstract" or "has full-text article" to find results that have a PubMed abstract or entire publication available.

Any number of search parameters may be added to refine a search, for example to construct a query for "paired-end RNA-Seq data from an Illumina HiSeq2500", but the return search must satisfy every search parameter. In other words, search parameters work as "A and B" and not as "A or B". In consequence, searches performed with two platform parameters for example "Platform = Illumina and Platform = Ion Torrent" will not return any results.

Note also that searches rely on metadata provided by the depositor of the SRA runs. This means that if, for example, an RNA-Seq run was not annotated as being RNA-Seq during submission, it won't be returned by a search for "Strategy = RNA-Seq".

Finally, the search tool uses NCBI's e-utilities, which occasionally experience downtime. If no searches return any results, check https://www.ncbi.nlm.nih.gov/sra/ to see the status of the service.

7.1.2 SRA search output

The search results are displayed with one run per line. Each Run Accession is a hyperlink to the NCBI webpage for the run, where additional information may be found, such as the distribution of nucleotide quality scores and links to external resources. On the right hand side of the search table, a "SRA Preview" panel shows the title and abstract associated to the selected run when available.

When looking for a specific run using the run SRA accession number, the tool will output the run that was searched and may also list additional runs that were submitted as part of the same experiment. In any case, it is safest to perform a new search specifically for the study to make sure the tool retrieves all possible runs. Right-click on a row to get a list of possible searches based on the selected run (figure 7.2), for example searching for more runs from the same sample, experiment or organism.

By default, the tool will output a maximum of 50 runs. The "more..." button below the table retrieves additional search results when the search exceeds 50 runs. The number of additional results returned can be controlled in Edit | Preferences | General | Number of hits (NCBI/Uniprot).

The "Total number of experiments" at the bottom of the search table reports how many experiments were retrieved, and not how many runs are listed in the table.

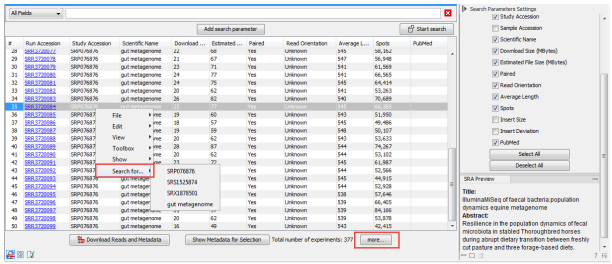


Figure 7.2: *The SRA search result table.*

At the bottom of the table, the button **"Show Metadata for Selection"** will create a metadata table containing sample specific information for the selected run(s). The first columns of the metadata table contain the same database identifiers as in the Search Table. The last columns recover sample details associated with the biosample. The list of all metadata available for the selected run is located in the right hand side panel of the table. This functionality is especially useful when selecting external data for use in a meta-analysis. For example, if your analysis is for a particular cancer type, and controls for patient age and gender, the **"Show Metadata for Selection"** facilitates the search for other datasets where the same metadata is available.

7.1.3 Downloading reads and metadata from SRA

Click on **Download Reads and Metadata** to save reads and their associated data. The data is saved in a metadata table and can be later associated to the reads for use in downstream analysis, for example to define factors for differential expression in the Differential Expression for RNA-Seq tool. Should the metadata table later be deleted, the "Show Metadata for Selection" button can be used to quickly recover a copy without having to re-download all the runs.

The Download Reads and Metadata wizard offers the following options:

Import Options (figure 7.3)

As with other NGS reads importers, it is possible to discard read names and/or quality scores to save space.

- "Download size" is the size of the .sra files that will be downloaded. Note that in some cases, the actual download may be up to 1GB larger than stated, as .sra files can be

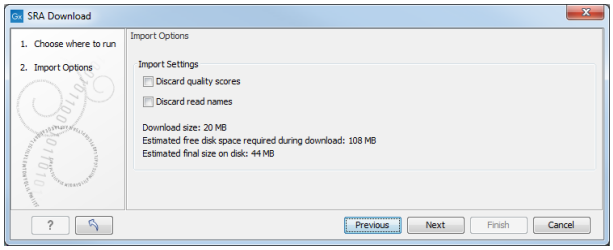


Figure 7.3: *The Download Reads and Metadata Import Options dialog.*

reference-compressed, meaning that a copy of the genome must also be retrieved before the file can be converted into fastq and imported into the workbench.

- "Estimated free disk space required during download" is a conservative estimate for the total free disk space required to download the selected runs. This is the "Estimated final size on disk" + the size of the largest single run in FASTQ format + the size of the largest single run in SRA format.
- "Estimated final size on disk" is an estimate of the total size of the files after they have been imported into the workbench.

Edit Paired End Settings (figure 7.4)

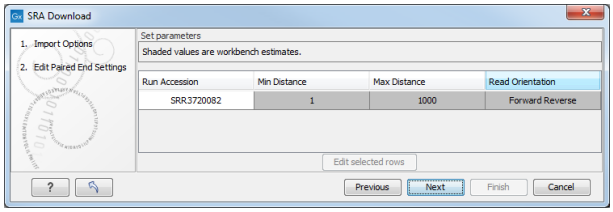


Figure 7.4: *The Download Reads and Metadata Edit Paired End Settings dialog.*

This dialog appears for all runs marked as being Paired (Paired column contains "Yes").

Read orientation is always guessed to be "Forward Reverse" unless otherwise stated.

Minimum distance and **Maximum distance** depend on how much data the depositor supplied with the runs. They are allowed to supply an "Insert Size" and an "Insert Deviation".

- If no insert size is supplied, we use defaults of 1 for minimum and 1,000 for maximum.
- If an insert size is supplied, we make the following calculation:
  - $Mindistance = insertsize - 5 * insertdeviation$
  - $Maxdistance = insertsize + 5 * insertdeviation$
- If no deviation is supplied, we estimate this to be $0.1 * insertsize$ and perform the same calculation as above.

When possible, we generally recommend that SRA data be used in subsequent analyses with the "Auto paired end distance detection" option enabled as the quality of deposited information is low. For example, some depositors report insert size including the length of the reads, and some excluding the length of the reads.

7.1.4 How reads are downloaded

SRA reads are downloaded in the ".sra" format using the NCBI SRA-toolkit. A .sra file is typically 2.5x smaller than an equivalent zipped fastq file. Download uses the NCBI 'prefetch' utility, and the resulting file is read into the workbench using 'fastq-dump'.

Sometimes runs in SRA cannot be downloaded. The affected runs are listed in a Problems panel together with a description of the problem. It is still possible to download the remaining runs.

The most common problems are:

- "The selected SRA reads contain no spots, and cannot be imported in the workbench.": The run has no associated sequencing data.
- "The selected SRA reads are dbGaP restricted.": For data protection reasons, you must request access to these reads. Requests and download cannot happen within the workbench, but you can follow the procedures here: http://www.ncbi.nlm.nih.gov/books/NBK5295/.
- "The selected SRA reads are made with an unsupported sequencing platform.": For example, Complete Genomics reads consist of eight regions separated by gaps of variable lengths, and should be analyzed by specialist tools.

We support download of reads via the commercial FASP protocol from Aspera. In our testing, Aspera download is up to 10x faster than a normal http download.

To enable this functionality, you have to download the Aspera Connect software from http://downloads.asperasoft.com/connect2/.

On Windows, choose to do a "Custom" install and choose to "Install for all users of this machine". You can then test if the installation worked by downloading a small file. The log (accessible from the Processes tab) will include the line "Downloading via fasp" if everything worked.

It is possible to change the Aspera options using **Preferences | Advanced | SRA Download**.

The following options are available:

- **Use Aspera when available** Per default, Aspera is automatically used if installed. This option makes it possible to disable Aspera.
- **Limit Aspera download speeds to [ ] Mb/s (Mac and Linux only)** Using Aspera may take up a lot of network resources. Use this option to specify a maximum download speed (in megabit per second). Note that this option is only available on Mac and Linux. For Windows users, it is possible to limit the maximum download speed by modifying the `aspera.conf file`, which can be found in `C:\Program Files (x86)\Aspera\Aspera Connect`. See http://download.asperasoft.com/download/docs/csrv/3.3.4/linux/html/index.html and http://download.asperasoft.com/download/docs/csrv/3.3.4/linux/html/fasp/setting-global-bandwidth.html for more details.

7.2 Sequence web info

Biomedical Genomics Workbench provides direct access to web-based search in various databases and on the Internet using your computer's default browser. You can look up a sequence in the

databases of NCBI and UniProt, search for a sequence on the Internet using Google and search for Pubmed references at NCBI. This is useful for quickly obtaining updated and additional information about a sequence.

The procedure for searching is identical for all four search options (see also figure 7.5):

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

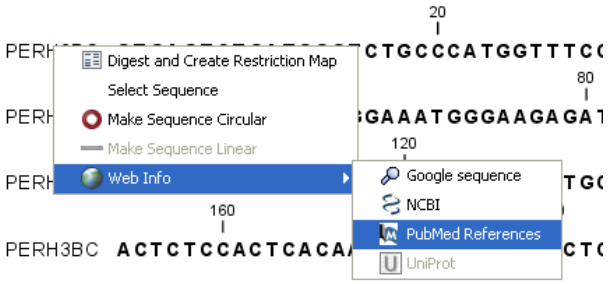


Figure 7.5: *Open webpages with information about this sequence.*

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

Google sequence The Google search function uses the accession number of the sequence which is used as search term on http://www.google.com. The resulting web page is equivalent to typing the accession number of the sequence into the search field on http://www.google.com.

NCBI The NCBI search function searches in GenBank at NCBI (http://www.ncbi.nlm.nih.gov) using an identification number (when you view the sequence as text it is the "GI" number). Therefore, the sequence file must contain this number in order to look it up at NCBI. All sequences downloaded from NCBI have this number.

PubMed References The PubMed references search option lets you look up Pubmed articles based on references contained in the sequence file (when you view the sequence as text it contains a number of "PUBMED" lines). Not all sequence have these PubMed references, but in this case you will se a dialog and the browser will not open.

UniProt The UniProt search function searches in the UniProt database (http://www.ebi.uniprot.org) using the accession number. Furthermore, it checks whether the sequence was indeed downloaded from UniProt.

Additional annotation information When sequences are downloaded from GenBank they often link to additional information on taxonomy, conserved domains etc. If such information is available for a sequence it is possible to access additional accurate online information. If the db_xref identifier line is found as part of the annotation information in the downloaded GenBank file, it is possible to easily look up additional information on the NCBI web-site.

To access this feature, simply right click an annotation and see which databases are available. For tracks, these links are also available in the track table.

Chapter 8

Running tools, handling results and batching

Contents

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

8.1 Running tools

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

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

You can open a tool from the Toolbox by double clicking on its name in the Toolbox. In case you do not know which folder the tool you are looking for belongs to, you can use the very useful Launch button () and type any part of the tool name in the search field. Double click on the name of the tool in the table.

When you open a tool, a wizard pop up in the center of the View Area. Through a succession of windows you will enter the data you want to analyze, the parameters of the analysis you want

to perform and how you want to handle the results of the analysis. You can navigate between windows by clicking the buttons **Next** and **Previous** at the bottom of the window.

If you are working on a network and have access to a server, you will first be asked to "Choose where to run" the tool. This window gives the following options:

- **Workbench** to run the tool on your own computer.
- **CLC Server** to run the tool on a server.
- **Grid** to be able to choose from the drop down menu.

If you check the option "Remember setting and skip this step", you will not have to enter the information described above for this particular tool. The setting can always be changed xxxx

After having decided where to run the tool, the next window of the wizard is usually asking you to select the input file(s). This window displays on the left a replicate of your Navigation Area in which only the files that have a format adapted to the tool will be shown. The specific input file formats required by a tool are described for each tool independently in the relevant sections of the manual that you can access easily by clicking on the **Help** button. For example, you can see a view of the Navigation Area in the workbench and the same Navigation Area in the wizard in figure 8.1. The Assemble Sequences tool will only accept nucleotide sequences (as such or part of a list), which explains why the file called "Read mapping", or the amino acid sequence ATP8a1 are not being displayed in the wizard Navigation Area.

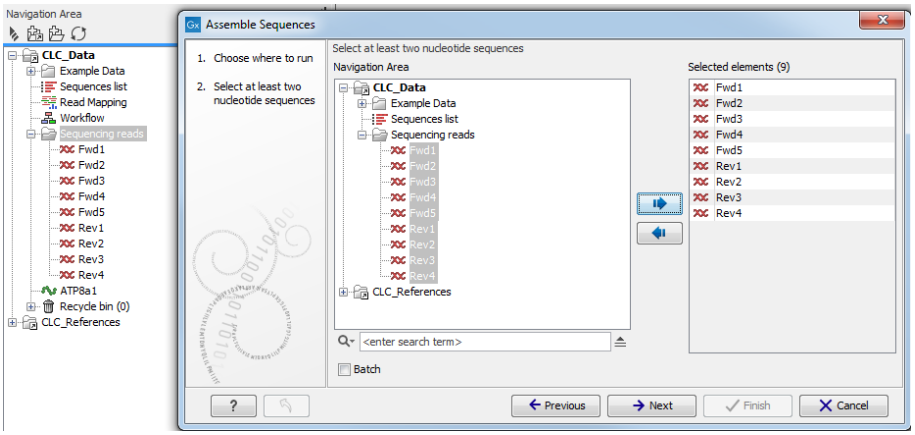


Figure 8.1: *You can select input files for the tool from the Navigation Area replicate on the left hand side of the wizard window.*

To select a file, you need to move it from the "Navigation Area" view to the "Selected elements" view. You can move a file by clicking on the arrows between the 2 views, or by double click on the file itself. Sometimes, having a file selected in the workbench Navigation Area will automatically put it in the Selected elements view. If you do not wish to work with this file as input, make sure to deselect it. You can deselect by using the arrow to send it out the selected elements view, or again by double clicking on it.

You also have the option in this window to work in batch, which means that the tool will run multiple times using each selected file as an independent dataset (as opposed to treat all selected files as a single input). To learn more about working in batch, please see section 8.3.

Once all elements are selected, you can click **Next** to proceed to the next step(s) of the wizard. During these steps you are usually required to set parameters for the tool (see figure 8.2 for

an example). You can read about specific settings for each tool in the relevant section of the manual accessible directly using the **Help** button. A pop up window will open to the section of the manual describing the tool you are using. The **Reset** button will reset all parameters from the pop window to their default values.

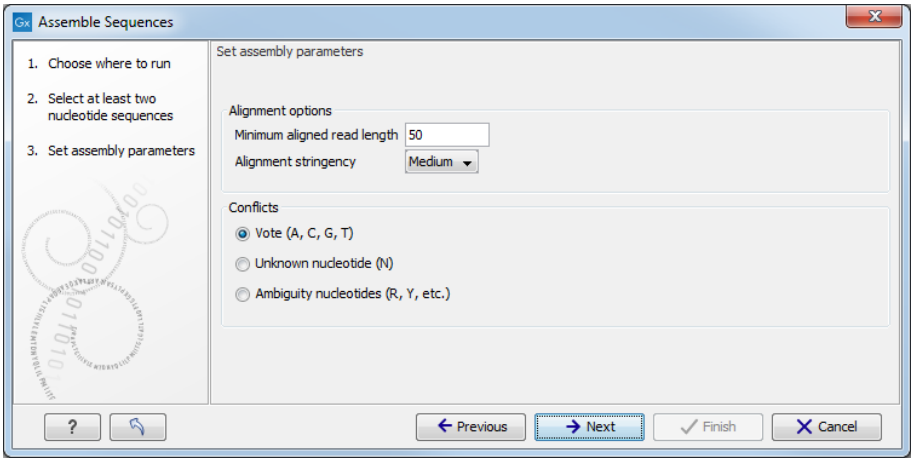


Figure 8.2: *An example of a "Set parameters" window.*

8.2 Handling results

A tool will output one or more result files, some of which are optional and can be selected - or deselected - in the last wizard window called "Result handling". The kind of output files generated by a tool as well as a description of additional files are described in the tool specific sections of the manual.

The "result handling" window also allows you to decide whether you want to open or save your results.

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

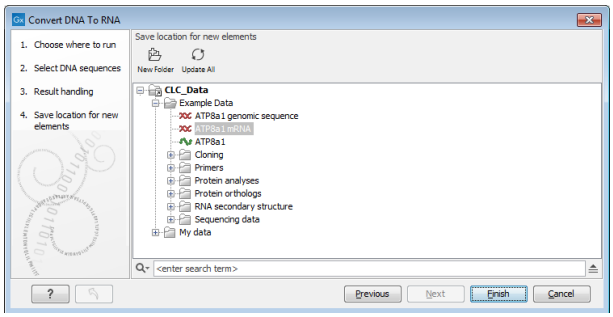


Figure 8.3: *Specify where to save the results of an analysis.*

You may also have the option to "Open log". A log is a textual view of the progress of the job.

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

If you chose "Open" results, they will open automatically open one or several Views in the View Area. Each View is described by a file name appended with an asterisk to indicate that this View has not been saved yet. To save it, drag the View tab to the relevant location in the Navigation Area, or simply use the usual Ctrl + S (or ⌘ + S). You can also right click on the tab and choose "Save" or use the "Save" button above the Navigation Area.

If you chose "Save" results, they will not open automatically, but they saved in the location you can specify in an extra wizard window. You can open the results by finding the file name in the Navigation Area after the tool is done processing, or by using the little arrow to the right of the analysis name in the Processes tab and choosing the option "Show results"(see figure 8.4).

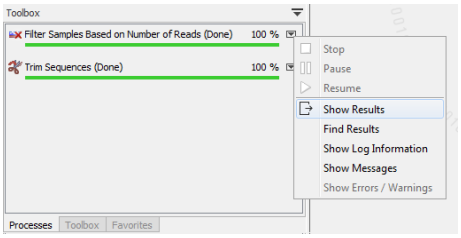


Figure 8.4: *Find your results using the little arrow to the right of the analysis name in the Processes tab.*

8.3 Batch processing

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

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

Batching installed workflows with multiple input elements, where **all** input elements will be changed per batch, is done differently (see section 8.3.5).

8.3.1 Standard batch processing

Standard batch mode is activated by clicking the **Batch** checkbox in the dialog where the input data is selected (figure 8.5).

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

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

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

- Each individual data element contained directly within the folder is a batch unit.

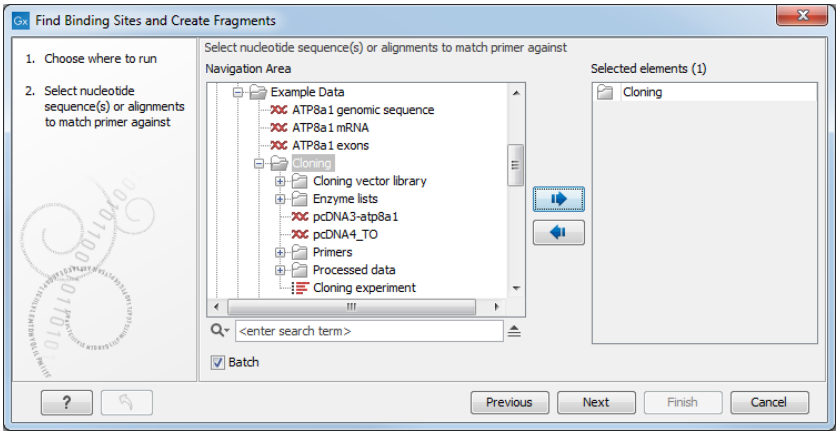


Figure 8.5: *The Cloning folder includes both folders and sequences.*

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

8.3.2 Batch overview

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

The `Cloning` folder that is found in the Example data (see section **??**) contains two sequences (🧬) and four folders (📁). If the Batch checkbox is checked and the Cloning folder is selected for an analysis, then after clicking on the button labeled **Next** an overview of the batch units like that in figure 8.6 is shown.

The batch overview lists the batch units on the left and the contents of the selected batch unit on the right.

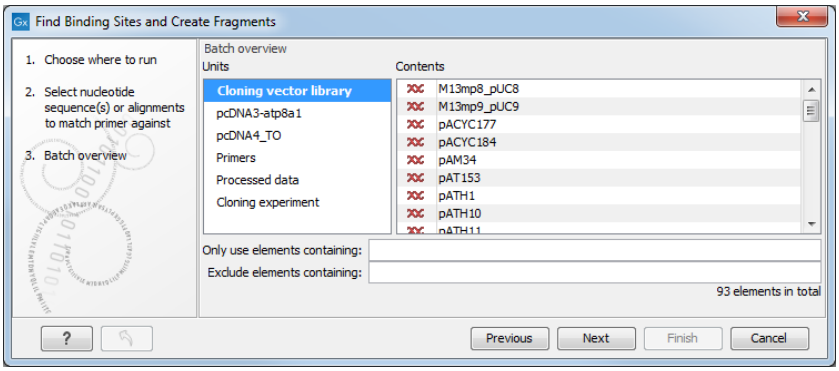


Figure 8.6: *Overview of the batch run. At the bottom right, the number of files to be analysed, summed across all batch units, is shown, 92 in this case.*

In this example, the two sequences (pcDNA) are defined as separate batch units because they are located at the top level of the Cloning folder. Of the four subfolders of the Cloning

folder initially selected, three are listed in this view. In each of these subfolders, any data elements that the analysis could use as input will be used unless action is taken at this point to exclude some of these. So, for example, in figure **??**, all the elements in the subfolder `Cloning vector library` and shown on the right-hand side of the dialog will be included as part of a single analysis run .

Note that the fourth subfolder of the Cloning folder, the `Enzyme lists` folder, is not listed as a batch unit. It is because it does not contain any data that can be used by the tool being launched.

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

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

8.3.3 Parameters for batch runs

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

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

8.3.4 Running the analysis and organizing the results

The last step in setting up a batch analysis is to choose where to save the outputs (figure 8.7).

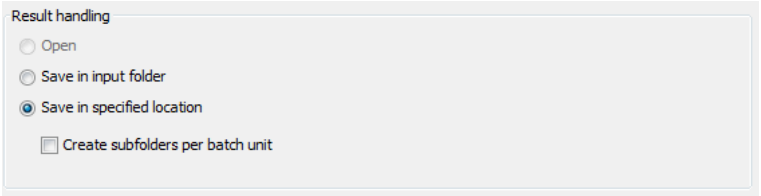


Figure 8.7: *Options for saving results when the tool was run in batch.*

The options available are:

- **Save in input folder** Save all outputs into the same folder as the input data. If the batch units consisted of folders, then the results of each analysis would be saved into the folder with the data it was generated using. If the batch units were individual data elements, then all the results will be placed into the same folder as those input data elements.
- **Save in specified location** Choose the folder where the outputs should be saved to, where when:
  - **Create subfolders per batch unit** is **unchecked**, all results for all batch units will be written to the specified folder.
  - **Create subfolders per batch unit** is **checked**, results for each batch unit will be written to a newly created subfolder of the selected folder. One subfolder is created per batch unit.

When the batch run is started, there will be one "master" process representing the overall batch job, and there will then be a separate process for each batch unit. The behavior this is different for Workbenches and Servers:

- On a Workbench, only one batch unit is run at a time. So when the first batch unit is done, the second will be started and so on. This avoids many parallel analyses that would draw on the same compute resources and slow down the computer.
- On a CLC Server, all the processes are placed in the queue, and the queue takes care of distributing the jobs. This means that if the server set-up includes multiple nodes, different batch unit analyses may be run in parallel.

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

For some analyses, there is an extra option in the final step to create a log of the batch process. This log will be created in the beginning of the process and continually updated with information about the results. The log will either be saved with the results of the analysis or opened in a view with the results, depending on how you chose to handle the results.

8.3.5 Batch launching workflows with multiple inputs

This section describes the launching of workflows with multiple inputs, where **all** input elements will be changed per batch. This launch mechanism is not intended for workflows with multiple input elements where one of the input elements remains the same in all batches, such as workflows meant to compare several tissues to a unique control tissue. At the moment, batch launching of such workflows is not possible, unless the common item is saved under different names as many times as there should be batches.

For workflows with multiple inputs where the inputs all need to change for each batch run, information specifying the grouping of the data elements and what role each element plays in a given analysis needs to be imported into the system from an Excel spreadsheet.

The requirements for launching such workflows in batch mode are:

- The workflow must be installed on the Workbench, meaning that the workflow is accessible from the Toolbox (as opposed to workflows accessible from the Navigation Area). See section 9.2 to learn how to install a workflow.
- The workflow is characterized by more than one input file, and all input elements are unique per batch. You cannot reuse a common input element (such as control reads for example), unless it has been saved under different names in the Navigation Area.
- An Excel format file (.xlsx/.xls) must be provided, with at least 3 different columns:
  - **Unique ID** The first column must contain either the exact name of the data elements to be used as inputs, or partial name information such that data elements being entered into the analysis can be uniquely identified and matched with the information contained in the spreadsheet (see section 3.2.3 to learn more about matching partial names).
  - **Grouping** A second column must specify which data elements should be analyzed together in a given batch unit: this would be the ID of a single individual when comparing different tissues from the same individual (one individual per batch); or a family name when identifying variants existing within one family (one family per batch).
  - **Type** The third column must specify the type for each data element: the values in this column distinguish tissue samples from controls, or inform about the disease status of a family member (affected/non-affected/proband) when identifying disease causing variants.

Ready-to-use workflows with more than one input in the *Biomedical Genomics Workbench* fall within two categories; 1) the Somatic Cancer workflow that compares tumor and normal samples, and 2) the Hereditary Disease workflows where a trio or a family of four is analyzed in one workflow.

(Figure 8.8) shows an example of the spreadsheet used in the Somatic Cancer workflows.

Unique ID = sample ID, exact of partial name of the reads file to ensure a unique match between reads and metadata.	Grouping = Identical values will be analyzed together in one batch unit, for example here Patient ID.	Type = value that defines which tissue is the control tissue and which is the sample tissue to be compared to the control.
23N	23	Normal
23T	23	Tumor
26N	26	Normal
26T	26	Tumor
27N	27	Normal
27T	27	Tumor
45N	45	Normal
45T	45	Tumor

Figure 8.8: *Example of a spreadhseet necessary to run a workflow in batch, where the workflow intend to compare two samples.*

To launch a workflow with multiple input elements in batch mode, right-click on the name of the workflow in the Toolbox and select the option "Run in Batch Mode..." (figure 8.9).

In the first dialog (figure 8.10) select the Excel file containing the information about the data to be analyzed should appear.

The data association table fills in with the data that is in the spreadsheet (figure 8.11).

Then specify the folder with the data, as shown in figure 8.12: click on the folder icon to the right and select the **folder** containing the data elements of interest. Subfolders and their contents are not considered unless the subfolder is also selected. Individual data elements cannot be selected.

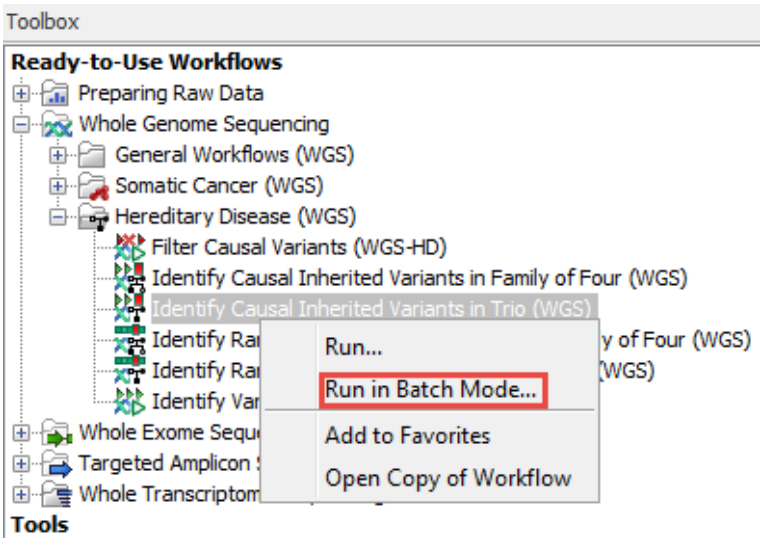


Figure 8.9: *The option to "Run in Batch Mode..." appears in the context menu when you right click on the name of an installed workflow that has multiple input elements in the Toolbox panel.*

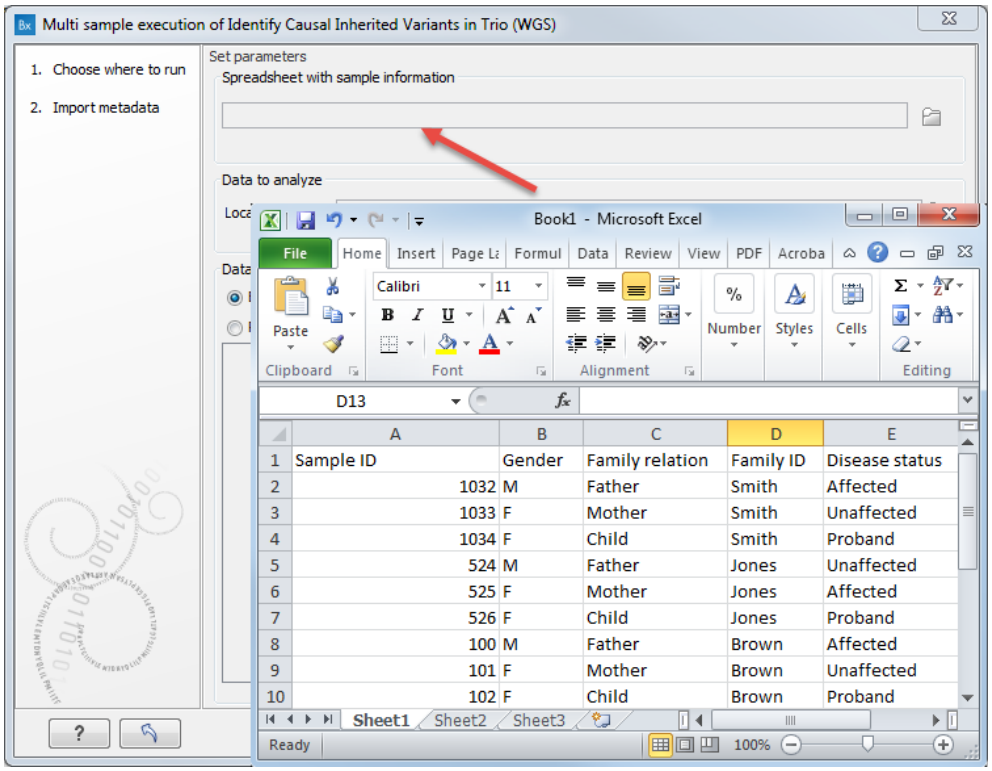


Figure 8.10: *Select the information about the data to be analyzed. An example of an Excel sheet with the relevant information is shown.*

Then select the appropriate matching scheme - exact or partial. The matching rules applied are the same as those used for metadata association. Exact means that data element names must exactly match an entry in the first column of the Excel file. Partial matching allows for data elements names to be only a part of an entry in the first column (but not the other way around). Partial matching rules are described in detail in section 3.2.3.

An icon with a green check mark (✓) appears in the table preview next to rows where a data element corresponding to a row of the Excel sheet was uniquely identified. If no match can be

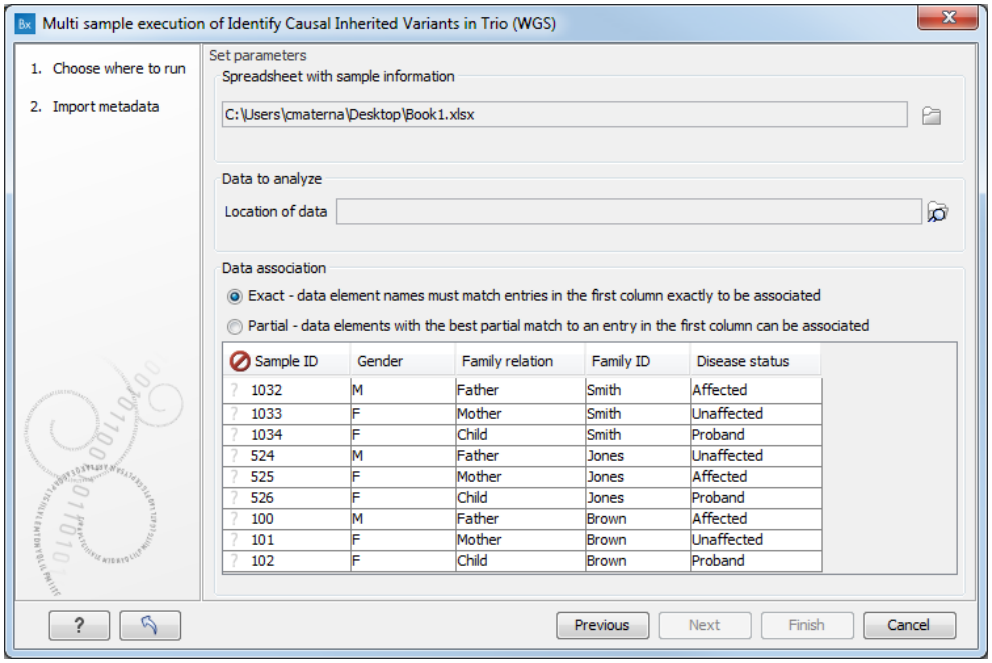


Figure 8.11: *When the Excel sheet has been selected, the table found in the lower part of the wizard will show the content of the Excel sheet. The location of the data for this analysis is not yet specified, so a red, no-entry sign is visible in the header of the first column.*

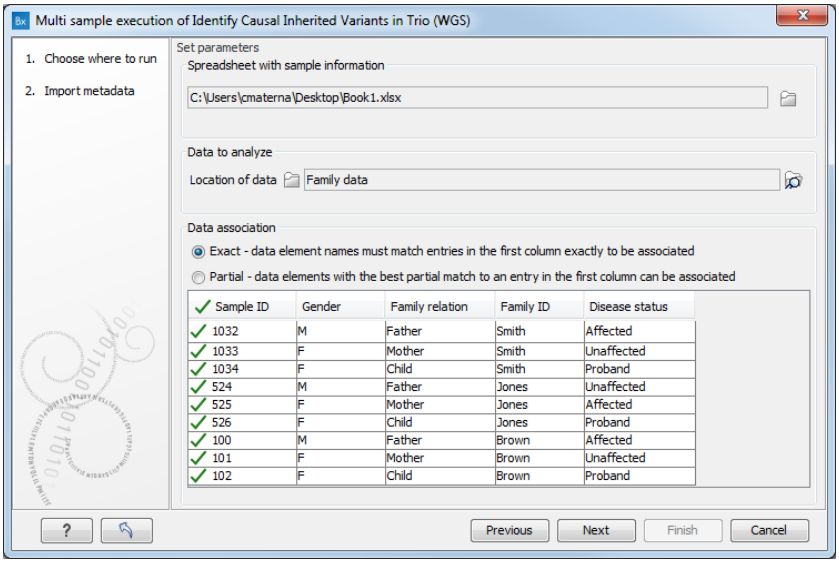


Figure 8.12: *Click on a folder or folders that contain the data to be analyzed. Here, the green check mark symbol in the header of the first column in the preview pane indicates that data elements were identified for each of the rows in the Excel sheet.*

made to a given row of the Excel sheet, a question mark (?) is displayed.

Graphical symbols are also presented in the header of the first column of the preview pane to give information about the overall status of the matching of rows in the Excel sheet with data elements in the Workbench:

- When no data elements match information in the Excel sheet, a red, no entry symbol ( ) is displayed. In this situation, the button **Next** is not enabled. This is the expected state

before any data elements have been selected.

- A yellow exclamation mark ( ) indicates that some, but not all rows in the Excel sheet have been matched to a data element in the selected folder(s).
- A green checkmark ( ) indicates that all rows in the Excel sheet have been matched to a data element in the selected folder(s).

In the next dialog (figure 8.13), define the following:

- In the **Group by** drop down menu, select the name of the column containing information that specifies which samples should be analyzed together. In the example, **Group by** is set to a column specifying family names, because each workflow run will analyze a particular family.
- In the **Type** drop down menu, select the name of the column containing information that can be mapped to the workflow input type of each data element. In the example, **Type** is set to the Disease status column, because the workflow inputs are an unaffected parent, an affected parent and a proband, and the Disease status column holds entries that can be mapped to these input types.

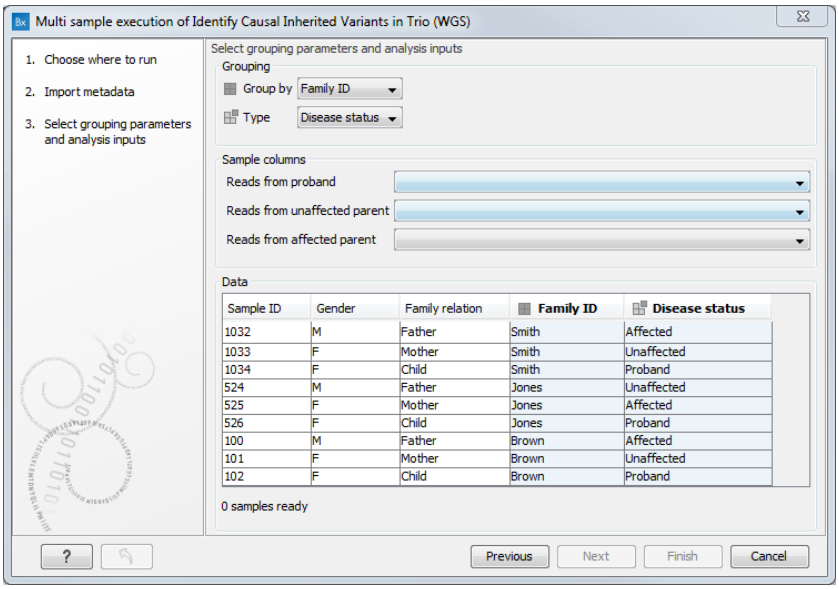


Figure 8.13: *A hereditary workflow is being lauched in batch mode. A given workflow run should analyze a family group, so the Group by entry is set to the column Family ID, where family groupings are specified. The workflow input types here are an unaffected parent, an affected parent and a proband. Information that can be mapped to these input types is held in the Disease status column, so this is selected in the Type drop down menu.*

Further details about the information in the Type column is now entered in the Sample columns area of the Wizard. For each input type for the workflow being launched, a drop down menu is provided containing the column entries from the column specified as containing the **Type** information.

Then, for each workflow input type listed, click on the drop down menu and select the term used to identify that particular input type (figure 8.14).

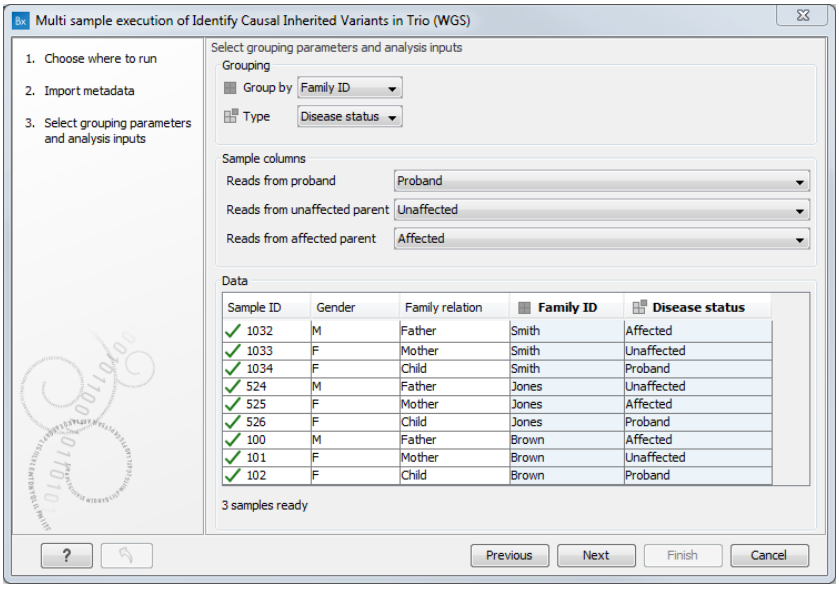


Figure 8.14: *The selections shown here indicate that data elements identified as matching rows from the Excel sheet containing "Proband" in the Disease status column should be used as the workflow input type "proband", data elements identified as matching rows containing "Unaffected" should be used as the workflow input type "unaffected parent", and data elements identified as matching rows containing "Affected" should be used as the workflow input type "affected parent".*

Click on the button labeled **Next** and work through any remaining Wizard steps where analysis details are presented and configure any unlocked parameters. Choose where to save the outputs of the analysis and click **Finish** to start the tool run in batches.

Important note: When running the Identify Rare Disease Causing Mutations ready-to-use workflows in batch mode, the gender of all proband samples in a given batch run must be the same. In other words, if multiple families are analyzed in a batch run, **the probands must all be female or they must all be male**. This is because proband gender is specified as a parameter, and the parameter values provided when setting up a workflow are then used for each analysis in the batch. The same condition applies when running a workflow in batch mode that includes a Trio Analysis. The gender of all child samples being analyzed in a given batch run must be the same.

Chapter 9

Workflows

Contents

The *Biomedical Genomics Workbench* provides a framework for creating, distributing, installing and running workflows. A workflow consists of a series of connected tools where the output of one tool is used as input for another tool. Once the workflow is set up, it can be installed (either in your own Workbench or it can be shared with colleagues and installed on a server). In that way it becomes possible to analyze lots of samples using the same standard pipeline, the same reference data and the same parameters.

This chapter will first explain how to create a new workflow, and next go into details about the installation and execution of a workflow.

Workflows created in the Workbench can also be installed on a *CLC Genomics Server*. For information about installing a workflow on the *CLC Genomics Server*, please see the user manual at http://www.qiagenbioinformatics.com/support/manuals/.

Note that the examples below are using tools from the *CLC Genomics Workbench* that are not necessarily available in the *Biomedical Genomics Workbench*. But the principles and workflow framework can be used in the same way with tools from *Biomedical Genomics Workbench*.

9.1 Creating a workflow

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

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

File | **New** | **Workflow ( )**

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

9.1.1 Adding workflow elements

First, click the **Add Element** ( ) button at the bottom (or use the shortcut Shift + Alt + E). This will bring up a dialog that lists the elements and tools, which can be added to a workflow (see figure 9.1).

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

Elements that can be selected in the dialog are mostly tools from the Toolbox. However, there are two special elements on the list: the elements that are used for input and output. These are explained in section 9.1.6.

You can select more than one element in the dialog by pressing Ctrl (⌘ on Mac) while selecting. Click OK when you have selected the relevant tools. You can add more later on if you wish.

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

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

9.1.2 Configuring workflow elements

Each of the tools can be configured by right-clicking the name of the tool as shown in figure 9.3.

The first option you are presented with is the option to **Rename** the element. This can be useful when you wish to discriminate between several copies of the same tool in a workflow. The name of the element is also visible as part of the process description when the workflow is run. To rename the element, right click on the tool in the workflow and select the "Rename" option, or

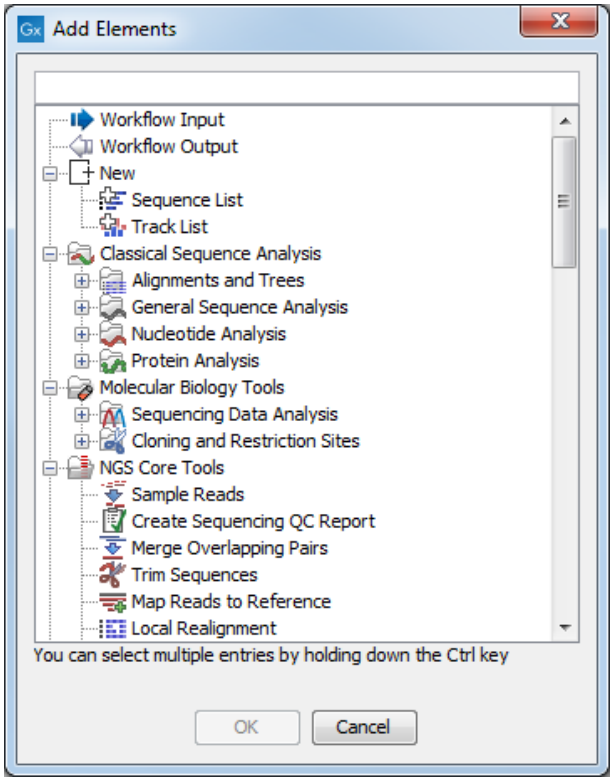


Figure 9.1: *Adding elements in the workflow.*

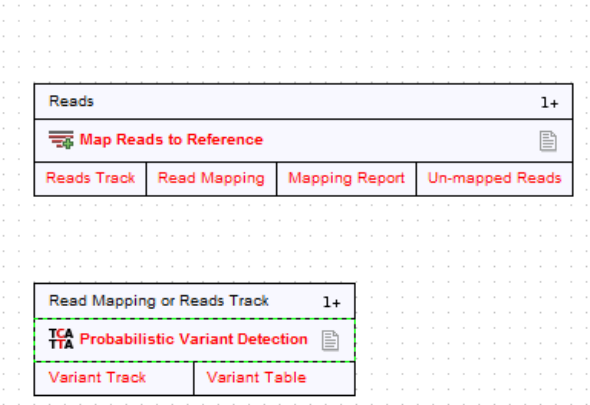


Figure 9.2: *Read mapping and variant calling added to the workflow.*

click on the tool in the workflow and then press the F2 key.

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

You can also configure a given element using the **Configure** option from the right click menu or by double-clicking on the element. This opens a dialog with options for setting parameters, selecting reference data, selecting the export destination of specified columns, etc. An example is shown in figure 9.4.

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

You can also change the name of a parameter if you so wish, for example, to help with usability

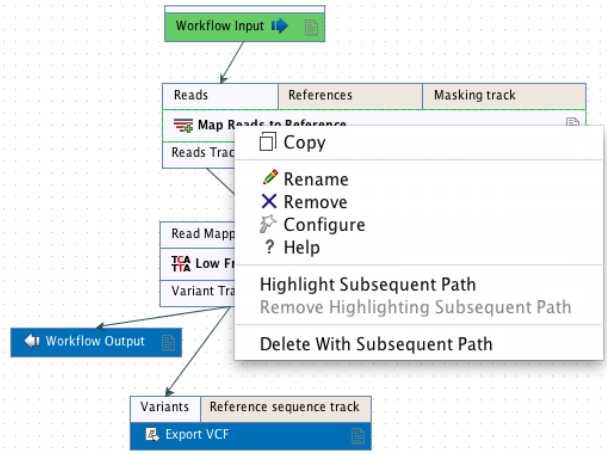


Figure 9.3: *Configuring a tool.*

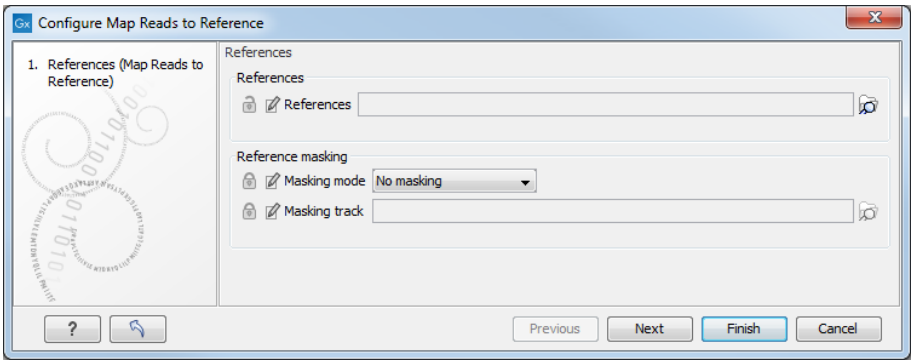


Figure 9.4: *Configuring read mapper parameters.*

for the intended users of a workflow. To do this, click on the edit icon ( ) and enter a new name.

Special consideration should be given when configuring reference data in a workflow. For example, when configuring a read mapping tool, such as shown in figure 9.3, you have to define a reference genome that sequences will be mapped to. You configure this by selecting data in the **Navigation Area**. If you distribute the workflow and it is installed on a different system, where that data is not accessible in same relative location, the workflow installation procedure will involve defining new reference data to use. This is explained in more detail in section 9.2.

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

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

The **Highlight Subsequent Path** option causes the element that was clicked on, and all the elements downstream of that one, to be highlighted. Other elements will be grayed out (figure 9.5). The **Remove Highlighting Subsequent Path** option reverts the highlighting, returning to the normal workflow layout.

In some workflows, many elements use the same reference data. There is a quick way of configuring these: right-click on the empty space and choose **Configure All References**. A dialog

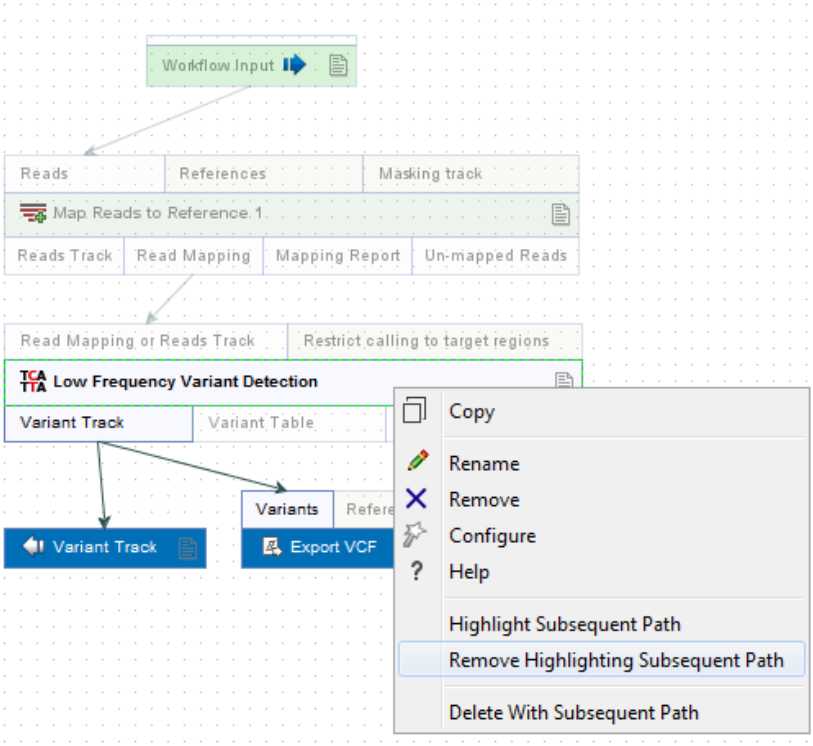


Figure 9.5: *Highlight path from the selected tool and downstream.*

then appears listing all the reference data needed by the workflow. When you click on the button labeled Finish, only the elements where the 'active' column is checked will be configured.

Similarly, instead of configuring the various tools individually, the **Configuration Editor** enables the specification of all settings, references, masking parameters etc. through a single wizard window (figure 9.6). This editor is accessed through the ( ) icon located in the lower left corner.

9.1.3 Locking and unlocking parameters

Figure 9.7 illustrates the different stages in the lifecycle of a workflow.

The first stage, workflow creation, was explained in the section above. The next stage, the installation of a workflow on a Workbench or Server is explained in section 9.2). The final stage is the executation of the workflow via the **Toolbox**, just like other tools.

During the creation step, the workflow author can specify which parameters should be locked or unlocked. If a parameter is locked, it cannot be changed in the installation or the execution step. Conversely, if it is left open, that parameter can be changed, either when running the Workflow or when installing it. See section 9.2). The lock icons shown in figure 9.4 specify whether the parameter is left open or whether it is locked.

By default, data parameters are unlocked. When installing the workflow on a different system to the one where it was created, the connection to the data needs to be re-established. This is only possible when the parameter is unlocked. Data parameters should only be locked if they should not be set, or if the workflow will only be installed in a setting where there is access to the same data in the same location as the system where the Workflow was created.

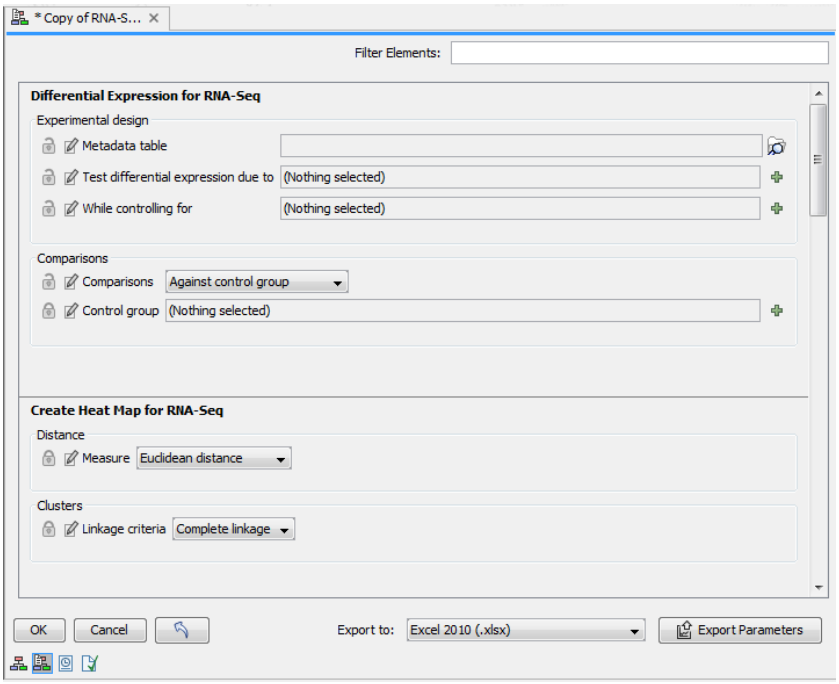


Figure 9.6: *The Configuration Editor can be used configure all the tools that can be configured in a given Workflow.*

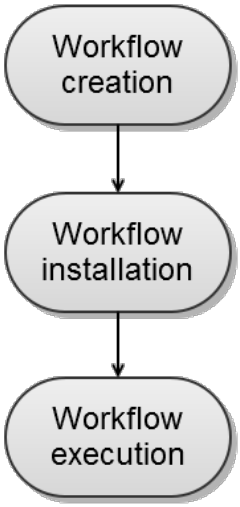


Figure 9.7: *The life cycle of a workflow.*

9.1.4 Connecting workflow elements

Figure 9.8 explains the different parts of a workflow element.

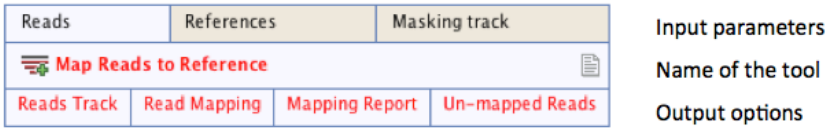


Figure 9.8: *A workflow element consists of three parts: input, name of the tool, and output.*

At the top of each element a description of the required type of input is found. In the right-hand

side, a symbol specifies whether the element accepts multiple incoming connections, e.g. +1 means that more than one output can be connected, and no symbol means that only one can be connected. At the bottom of each element there are a number of small boxes that represent the different kinds of output that is produced. In the example with the read mapper shown in figure 9.2, the read mapper is able to produce a reads track, a report etc.

Each of the output boxes can be connected to further analysis in three ways:

- By dragging with the mouse from the output into the input box of the next element. This is shown in figure 9.9. A green border around the box will tell you when the mouse button can be released, and an arrow will connect the two elements (see figure 9.10).
- Right-clicking the output box will display a list of the possible elements that this output could be connected to. You can also right-click the input box of an element and connect this to a matching output of another element.
- Alternatively, if the element to connect to is not already added, you can right-click the output and choose **Add Element to be Connected**. This will bring up the dialog from figure 9.1, but only showing the tools that accepts this particular output. Selecting a tool will both add it to the workflow and connect with the output you selected. You can also add an upstream element of workflow in the same way by right-clicking the input box.

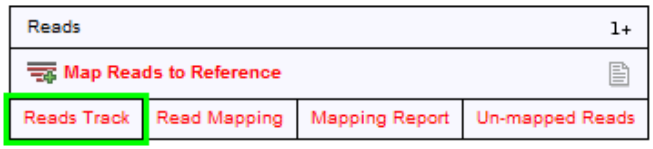


Figure 9.9: *Dragging the reads track output with the mouse.*

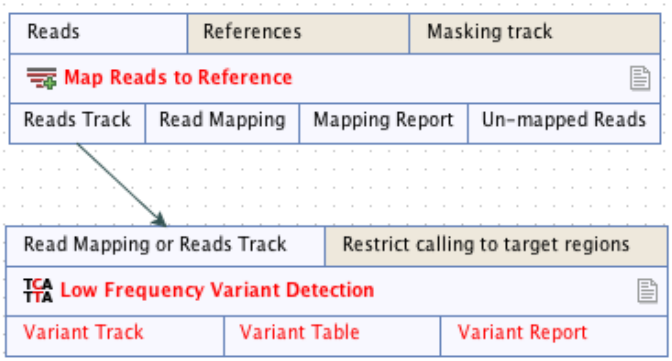


Figure 9.10: *The reads track is now used for variant calling.*

All the logic of combining output and input is based on matching the type of input. So the read mapper creates a reads track and a report as output. The variant caller accepts reads tracks as input but not mapping reports. This means that you will not be able to connect the mapping report to the variant caller.

Figure 9.11 demonstrates how one tool can receive input from two different sources; 1) a reads track that is the input that hold the data that is to be analyzed (in this case reads that is to be locally realigned), and 2) a parameter that can have different functions depending on the tool

that it is connected to (in this case the InDel track is used as a guidance track for the local realignment. In other situations the parameter track could be used for e.g. annotation or could provide a reference sequence).

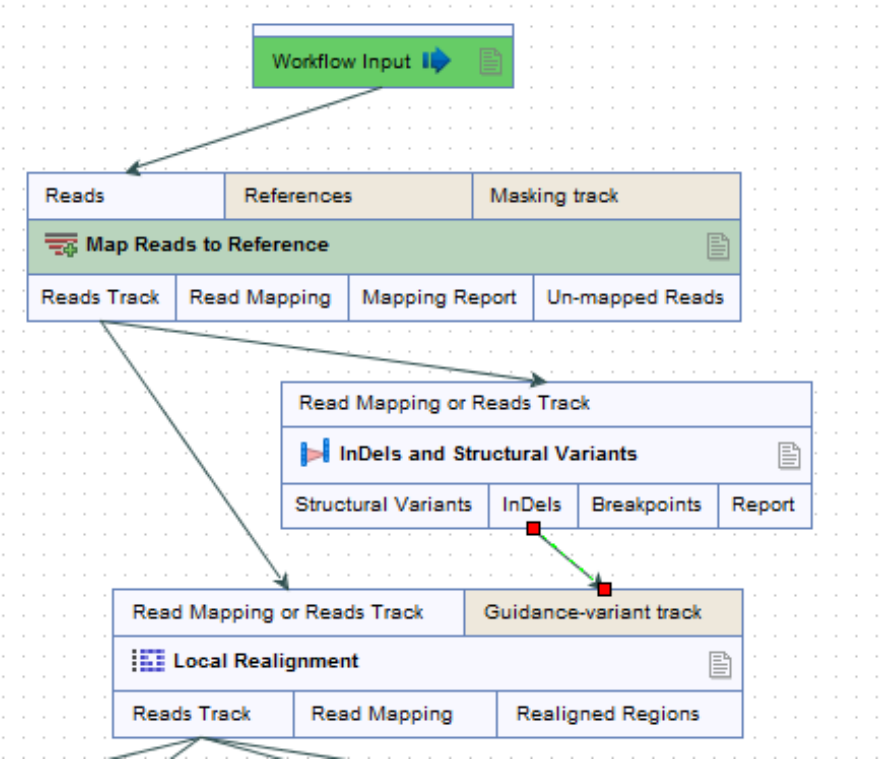


Figure 9.11: *A tool can receive input from both the generated output from another tool (in this example a reads track) and from a parameter (in this case indels detected with the InDels and Structural Variants tool).*

9.1.5 Output

Besides connecting the elements together, you have to decide what the input and the output of the workflow should be. We will first look at specification of the output, which is done by right-clicking the output box of any tool and selecting **Use as Workflow Output** as shown in figure 9.12.

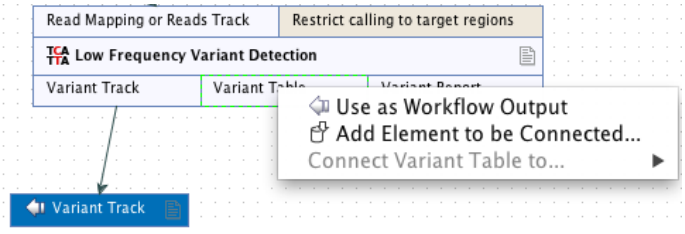


Figure 9.12: *Selecting a workflow output.*

You can mark several outputs this way throughout the workflow. Note that no intermediate results are saved unless they are marked as workflow output. In fact, when the workflow is executed, all

the intermediate results are indeed saved temporarily, but they are automatically deleted when the workflow is completed. However, if a workflow fails, the intermediate results are not deleted and will be found in a folder named after the workflow with the mention "intermediate".

By double-clicking the output box, you can specify how the result should be named as shown in figure 9.13.

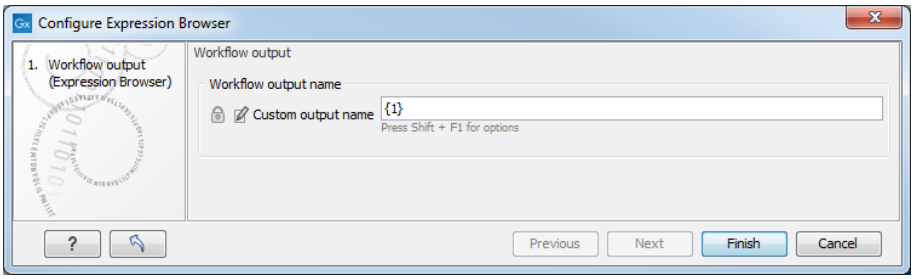


Figure 9.13: *Specifying naming of a workflow output.*

In this dialog you can enter a name for the output. Dynamic placeholders are available, which can help in setting up specific and standardized names for outputs. If you mouse over the Custom output name field in the dialog, the placeholders are listed. You can also click on Shift+F1 to see the options. The placeholdersi below are available. They are not case specific.

- **{name}** or **{1}** - default name for the tool's output
- **{input}** or **{2}** - the name of the workflow input (and not the input to a particular tool within a workflow).
- **{user}** - name of the user who launched the job
- **{host}** - name of the machine the job is run on
- **{year}**, **{month}**, **{day}**, **{hour}**, **{minute}**, and **{second}** - timestamp information based on the time an output is created. Using these placeholders, items generated by a workflow at different times can have different filenames.

When deciding on an output name, you can choose any combination of the placeholders, as well as custom names and punctuation, for example, `{input}({day}-{month}-{year})`. A meaningful name to a variant track could be `{2} variant track` as shown in figure 9.14. Here, if your workflow input was named `Sample 1`, the result would be "Sample 1 variant track".

The placeholders available for exports are slightly different than for other workflow outputs and are described in section 6.6.

It is also possible to save workflow outputs into subfolders by using a forward slash character `/` at the start of the output name definition. For example the custom output name `/variants/{name}` refers to a folder "variants" that would lie under the location selected for storing the workflow outputs. When defining subfolders for outputs, all later forward slash characters in the configuration, except the last one, will be interpreted as further levels of subfolders. For example, a name like `/variants/level2/level3/myoutput` would put the data item called `myoutput` into a folder called `level3` within a folder called `level2`, which itself is inside a folder called `variants`. The `variants` folder would be placed under the

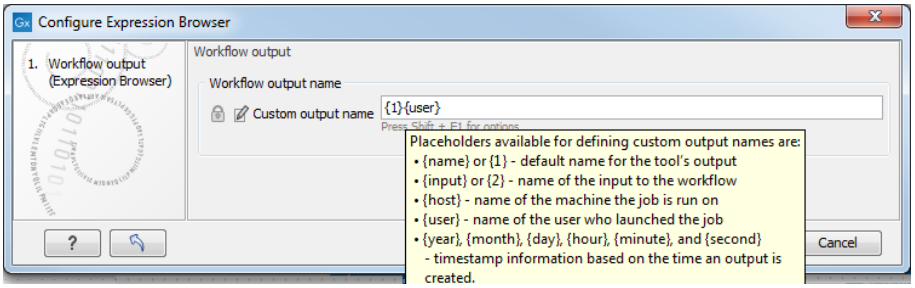


Figure 9.14: *Providing a custom name for the result.*

location selected for storing the workflow outputs. If the folders specified in the configuration do not already exist, they are created.

Exports are different to other workflow outputs in this regard; subfolders cannot be defined using the Custom file name field. If slash characters are included in the Custom file name field for an export, all text before and including the final slash character is ignored.

9.1.6 Input

In addition to output, you also have to specify where the data should go into the workflow by adding an element called **Workflow Input**. This can be done by:

- Right-clicking the input box of the first tool and choosing **Connect to Workflow Input**. By dragging from the workflow input box to other input boxes several tools can use the input data directly.
- Pressing the button labeled **Add Element** (or right-click somewhere in the workflow background area and select **Add Element** from the menu that appears). The input box must then be connected to the relevant tool(s) in the workflow by dragging from the Workflow Input box to the "input description" part of the relevant tool(s) in the workflow.

At this point you have only prepared the workflow for receiving input data, but not specified which data to use as input. To be able to do this you must first save the workflow. When this has been done, the button labeled **Run** is enabled which allows you to start executing the workflow. When you click on the button labeled **Run** you will be asked to provide the input data.

Multiple input files can be used when:

- Data is generated within the Workflow
- Data is held within the Workbench
- Data is a combination of the two situations above

It can be useful to rename input elements when working with multiple input files, so that it is easy to discriminate between them when they are shown during workflow execution.

Note: Once the multiple input feature is used in a workflow, it is not possible to run the workflow in batch mode.

You can choose the order in which inputs will be processed by an element by right clicking on the input parameter box at the top of the element and choosing the option **Order Inputs**. This is most relevant for elements involved in data visualization. The feature **Order Inputs** is enabled when there are at least two inputs connected to the element (see figure 9.15). A small window will open, in which you can indicate the preferred order of the inputs to that element by moving them up and down in the list (figure 9.16). From this point forward, the order of the inputs is displayed on the branches connecting the inputs to elements.

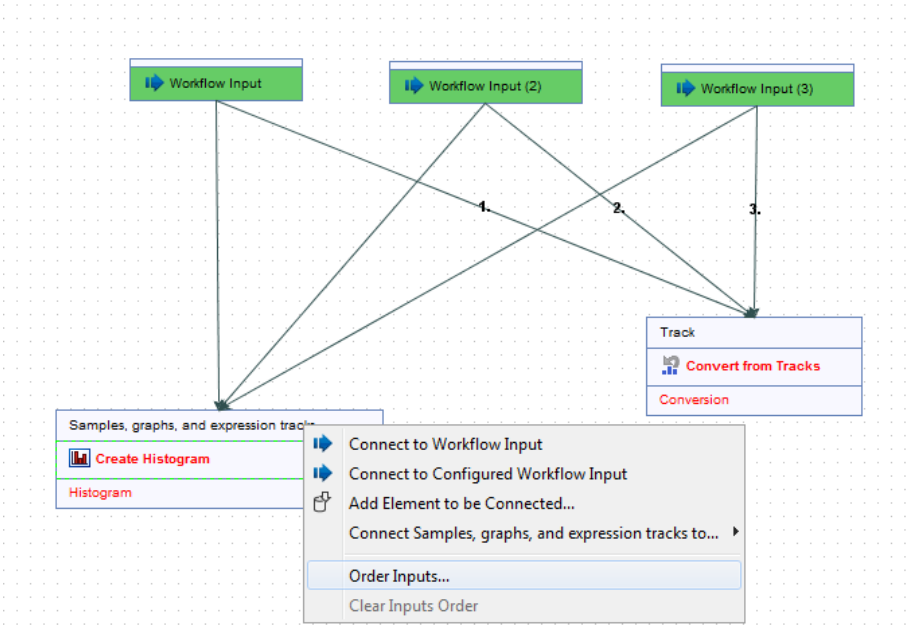


Figure 9.15: *Right-click on the input parameter box to see the Order Inputs function.*

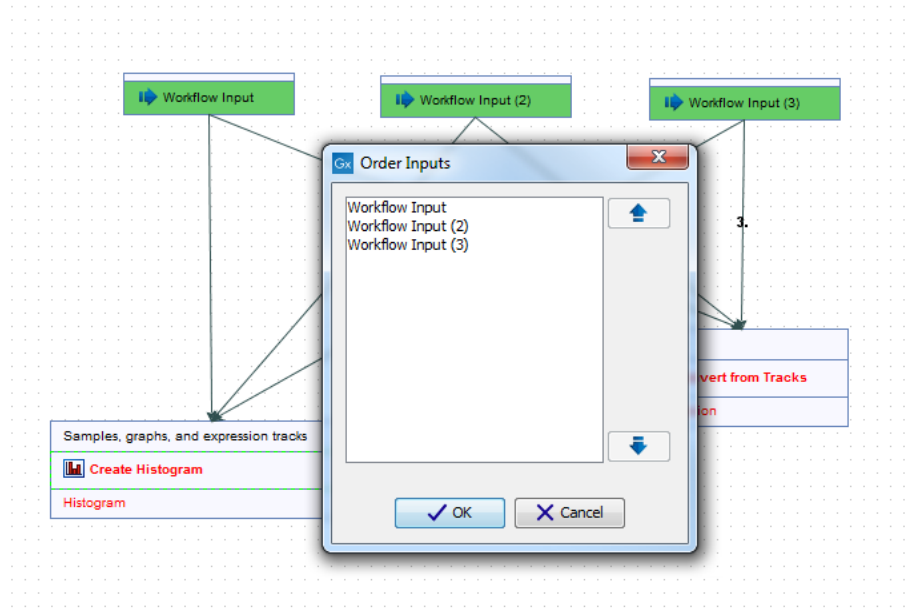


Figure 9.16: *Define the inputs order for the element.*

The feature **Order Workflow Inputs** allows you to set the order that a user will be asked for each input when they run the workflow. This option is enabled as soon as the workflow has two or more inputs (figure 9.17). Right click on empty space in the Workflow editor to start this tool. A small window will open in which the different inputs can be moved up and down to indicate the desired order (figure 9.18).

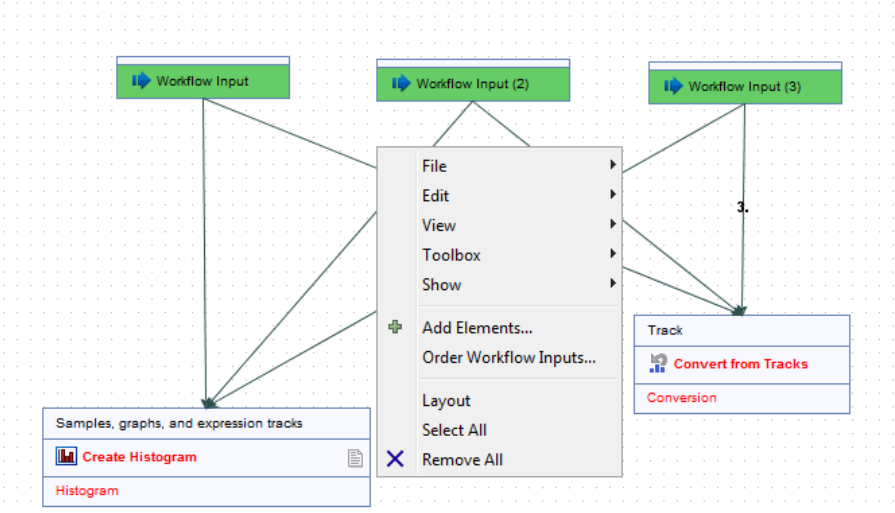


Figure 9.17: *Right click on empty space in the Workflow editor to open the Order Workflow Inputs tool.*

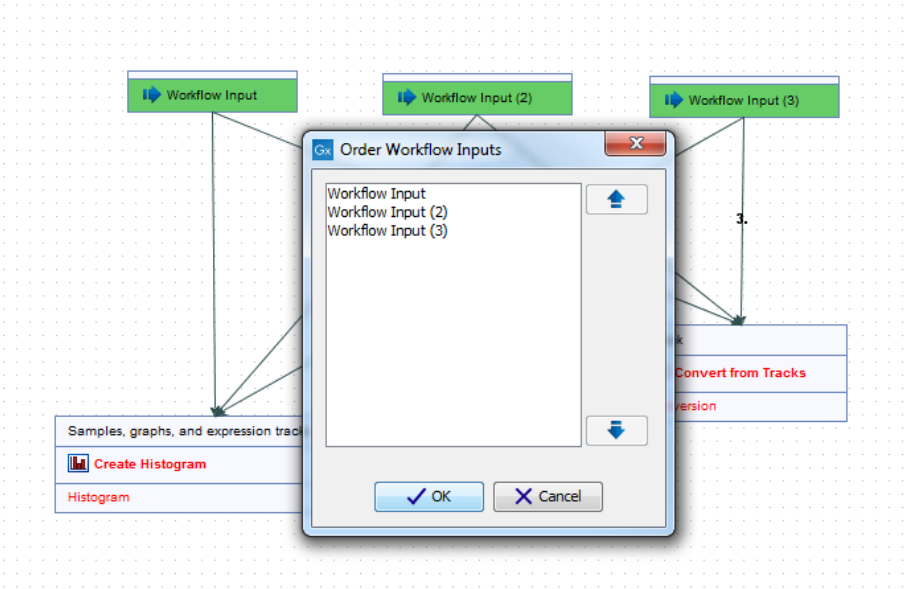


Figure 9.18: *Define the order of the inputs for the workflow.*

The example in figure 9.19 shows how to generate a track list in a workflow. Any track based on a compatible genome can be added to the same track list. This includes reference tracks as well as track results generated by elements of that workflow. In the latter case, only those for which a workflow output element has been configured can be included in a track list.

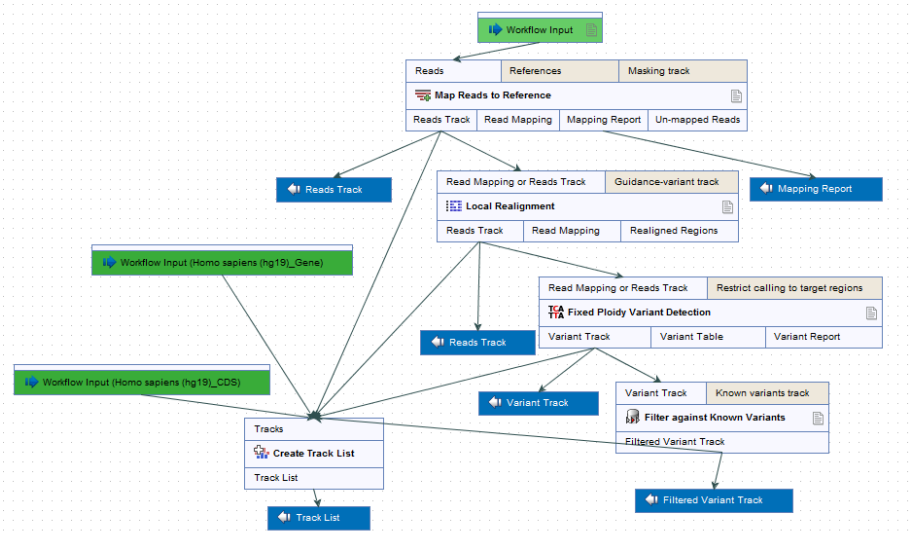


Figure 9.19: *Generation of a track list including data generated within the Workflow, as well as data held in the Workbench.*

9.1.7 Layout

The workflow layout can be adjusted automatically. Right clicking in the workflow editor will bring up a pop-up menu with the option "Layout". Click on "Layout" to adjust the layout of the selected elements (Figure 9.20). Only elements that have been connected will be adjusted.

Note! The layout can also be adjusted with the quick command Shift + Alt + L.

Note! It is very easy to make an image of the workflow. Simply select the elements in the workflow (this can be done pressing Ctrl + A, by dragging the mouse around the workflow while holding down the left mouse button, or by right clicking in the editor and then selecting "Select All"), then press the Copy button in the toolbar ( ) or CTRL + C. Press Ctrl + V to paste the image into the wanted destination e.g. an email or a text or presentation program.

9.1.8 Input modifying tools

An input modifying tool is a tool that manipulates its input objects (e.g. adds annotations) without producing a new object. This behavior differs from the rest of the tools and requires special handling in the workflow.

In the workflow an input modifying tool is marked with the symbol ( ) (figure 9.21).

Restrictions apply to workflows that contain input modifying tools. For example, branches are not allowed where one of the elements is a modifying tool (see figure 9.22), as it cannot be guaranteed which workflow branch will be executed first, which in turn means that different runs can result in production of different objects. Hence, if a workflow is constructed with a branch where one of the succeeding elements is a modifying tool, a message in red letters will appear saying "Branching before a modifying tool can lead to non-deterministic behavior". In such a situation the "Run" and "Create Installer" buttons will be disabled (figure 9.22).

The problem can be solved by resolving the branch by putting the elements in the right order (with

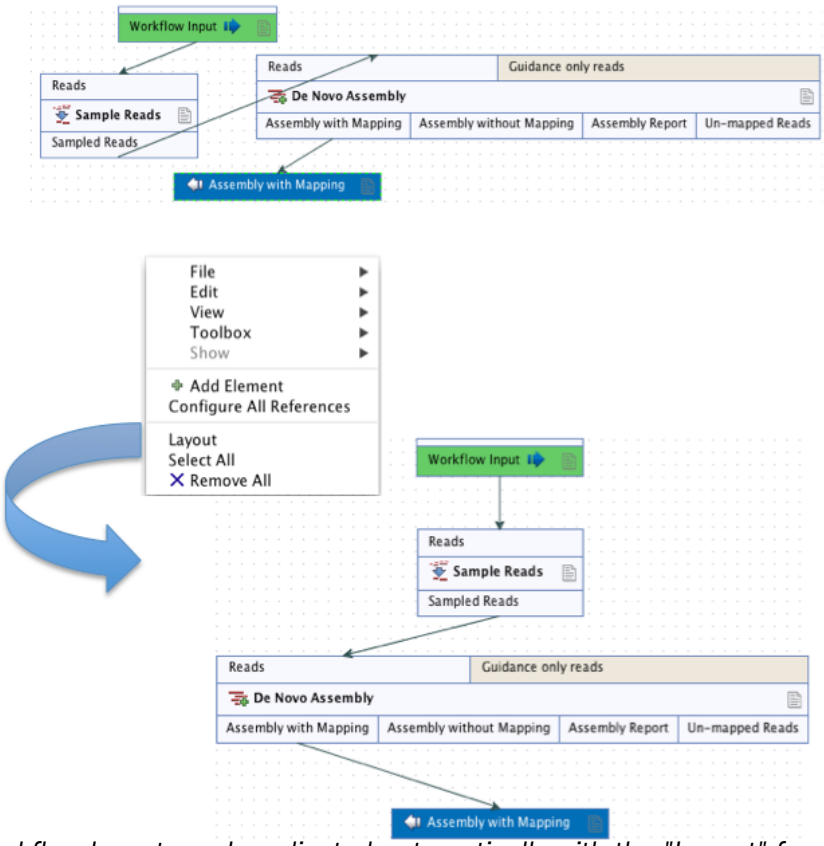


Figure 9.20: *A workflow layout can be adjusted automatically with the "Layout" function.*

Figure 9.21: *Input modifying tools are marked with the letter M.*

respect to order of execution). This is shown in figure 9.23 that also shows that the "Run" and "Create Installer" buttons are now enabled. In addition, a message in green letters has appeared saying "Validation successful".

As input modifying tools only modify existing objects without producing a new object, it is not possible to add a workflow output element directly after an input modifying tool (figure 9.24). A workflow output element can only be added when other tools than input modifying tools are included in the workflow.

If the situation occur where more input modifying tools are used succeedingly, a copy of the object will be created in addition to using the modified object as input at the next step of the chain (see figure 9.25). In order to see this output you must right click on the output option (marked with a red arrow in figure 9.25) and select "Use as Workflow Output".

When running a workflow where a workflow output has been added after the first input modifying

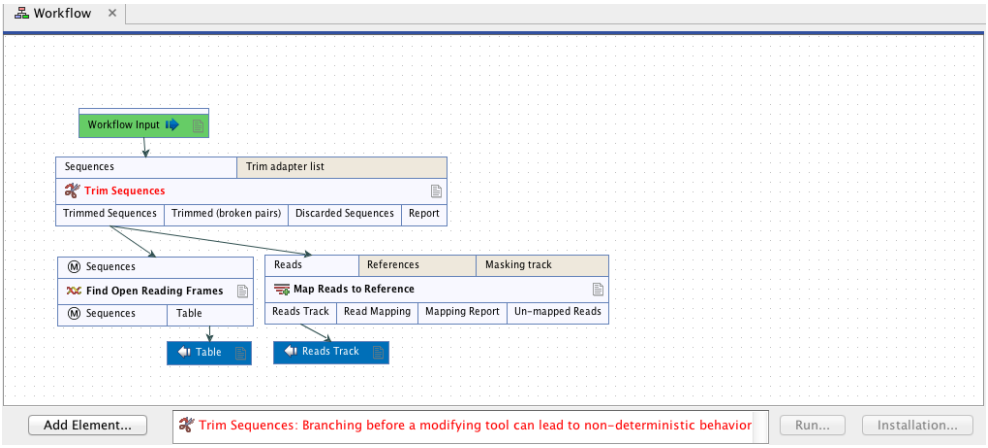


Figure 9.22: *A branch containing an input modifying tool is not allowed in a workflow.*

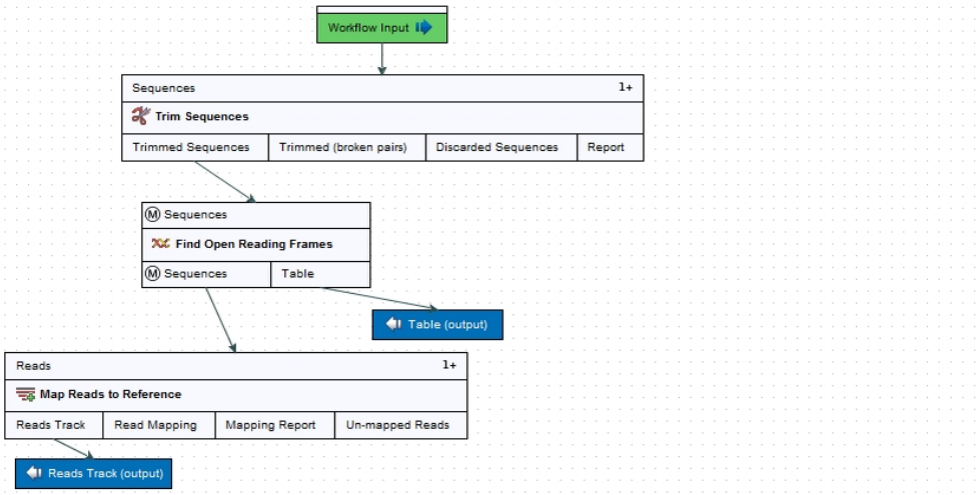


Figure 9.23: *A branch containing an input modifying tool has been resolved and the workflow can now be run or installed.*

tool in the chain (see figure 9.26) the output arrow is marked with "copy" to indicate that this is a copy of the result that is used as input at the next level in the chain. When running this workflow you will be able to see the copy of the output from the first input modifying tool in the **Navigation Area** (at the destination that you selected when running the workflow).

9.1.9 Workflow validation

At the bottom of the view, there is a text with a status of the workflow (see figure 9.27). It will inform about the actions you need to take to finalize the workflow.

The validation may contain several lines of text. Scroll the list to see more lines. If one of the errors pertain to a specific element in the workflow, clicking the error will highlight this element.

The following needs to be in place before a workflow can be executed:

- All input boxes need to be connected either to the workflow input or to the output of other

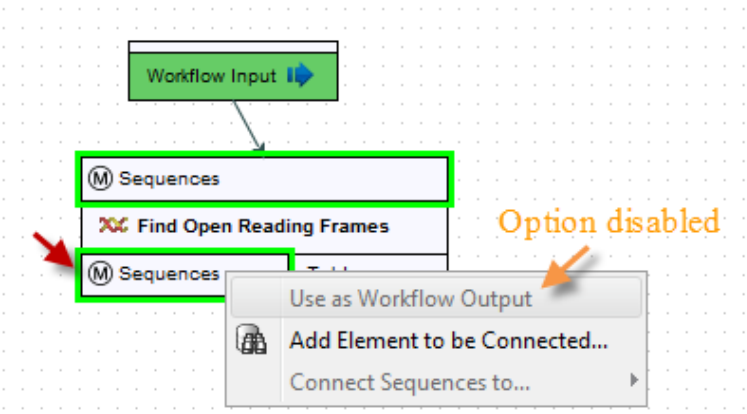


Figure 9.24: *A workflow output element cannot be added if the workflow only contains an input modifying tool.*

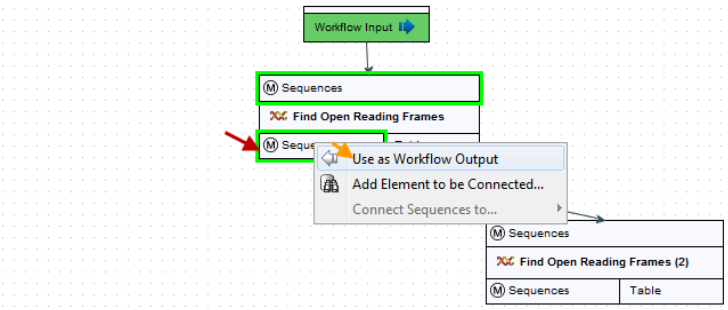


Figure 9.25: *A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Select "Use as Workflow Output" to make a copy of the output.*

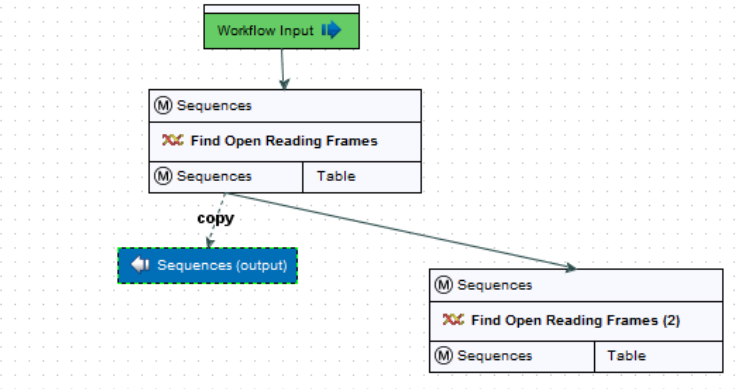


Figure 9.26: *A workflow output element can be added when more than one input modifying tool is used succeedingly (despite that the workflow only contains input modifying tools). Note that this output is marked with "copy" to indicate that this is a copy of the result that is used as input at the next level in the chain.*

tools.

- At least one output box from each tool needs to be connected to either a workflow output or to the input box of another tool.
- Additional checks that the workflow is consistent.

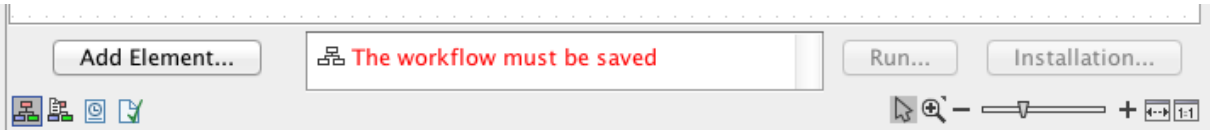


Figure 9.27: *A workflow is constantly validated at the bottom of the view.*

Once these conditions are fulfilled, the status will be "Validation successful", the **Run** button is enabled. Clicking this button will enable you to try running a data set through the workflow to test that it produces the expected results. If reference data has not been configured (see section 9.1.2), there will be a dialog asking for this as part of the test run.

9.1.10 Workflow creation helper tools

In the workflow editor **Side Panel**, you will find the following workflow display settings that can be useful to know (figure 9.28):

Grid

- Enable grid You can display a grid and control the spacing and color of the grid. Per default, the grid is shown, and the workflow elements snap to the grid when they are moved around.

View mode

- Collapsed The elements of the workflow can be collapsed to allow a cleaner view and especially for large workflows this can be useful.
- Highlight used elements Ticking **Highlight used elements** (or using the shortcut Alt + Shift + U) will show all elements that are used in the workflow whereas unused elements are grayed out.
- Rulers Vertical and horizontal rules can be visualised
- Auto Layout Ticking **Auto Layout** will ensure rearrangement of elements once new elements are added.
- Connections to background Connecting arrows are shown behind elements. This may easy reading of element names and accessible parameters.

Design

- Round elements Enable rounding of the element boxes.
- Show shadow Shadows of element boxes can be added.
- Configured elements Background color can be customized.
- Inpupt elements Background color can be customized.
- Edges Color of connecting arrows can be customized.

9.1.11 Adding to workflows

Additional elements can be added to an already existing workflow by dragging it from the navigation area into the workflow editor and joining more elements as necessary. The new

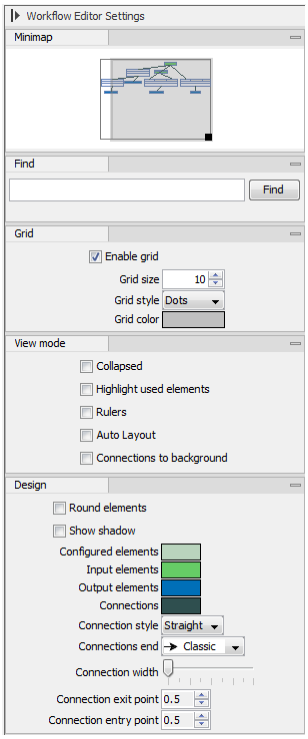


Figure 9.28: *The Side Panel of the workflow editor.*

workflow must be saved and validated before it can be executed. Two or more workflows can be joined by dragging and dropping one from the Navigation Area, into another that is already open in the main viewing area. The output of one must be connected to the input of the next to allow the whole workflow to run in one go.

Workflows do not need to be valid to be dragged in to the workflow editor, but they must have been updated to the current version of the workbench.

9.1.12 Snippets in workflows

When creating a new workflow, you will often have a number of connected elements that are shared between workflows. Instead of building workflows from scratch it is possible to reuse components of an existing workflow. These components are called snippets and can exist of e.g. a read mapper and a variant caller.

Snippets can be created from an existing workflow by selecting the elements and the arrows connecting the selected elements. Next, you must right-click in the center of one of the selected elements. This will bring up the menu shown in figure 9.29.

When you have clicked on "Install as snippet" the dialog shown in figure 9.30 will appear. The dialog allows you to name the snippet and view the selected elements that are included in the snippet. You are also asked to specify whether or not you want to include the configuration of the selected elements and save it in the snippet or to only save the elements in their default configuration.

Click on the button labeled **OK**. This will install your snippet and the installed snippet will now appear in the **Side Panel** under the "Snippets" tab (see figure 9.31)

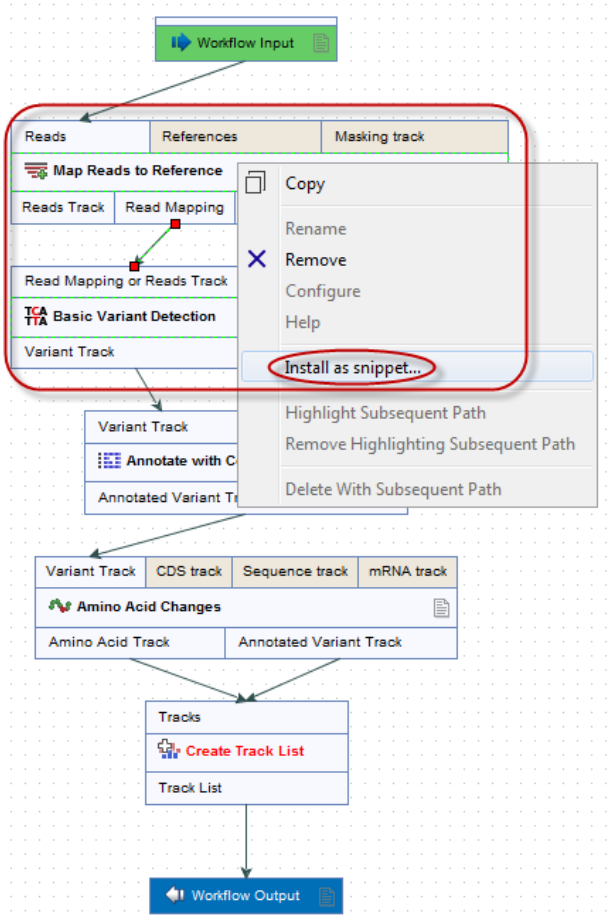


Figure 9.29: *The selected elements are highlighted with a red box in this figure. Select "Install as snippet".*

Right-clicking on the installed snippet in the **Side Panel** will bring up the following options (figure 9.32):

- **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.
- **Update** Updates the snippet (if update is required).

If you right-click on the top-level folder you get the options shown in figure 9.33:

- **Create new group** Creates a new folder under the selected folder.
- **Remove group** Removes the selected group (not available for the top-level folder)

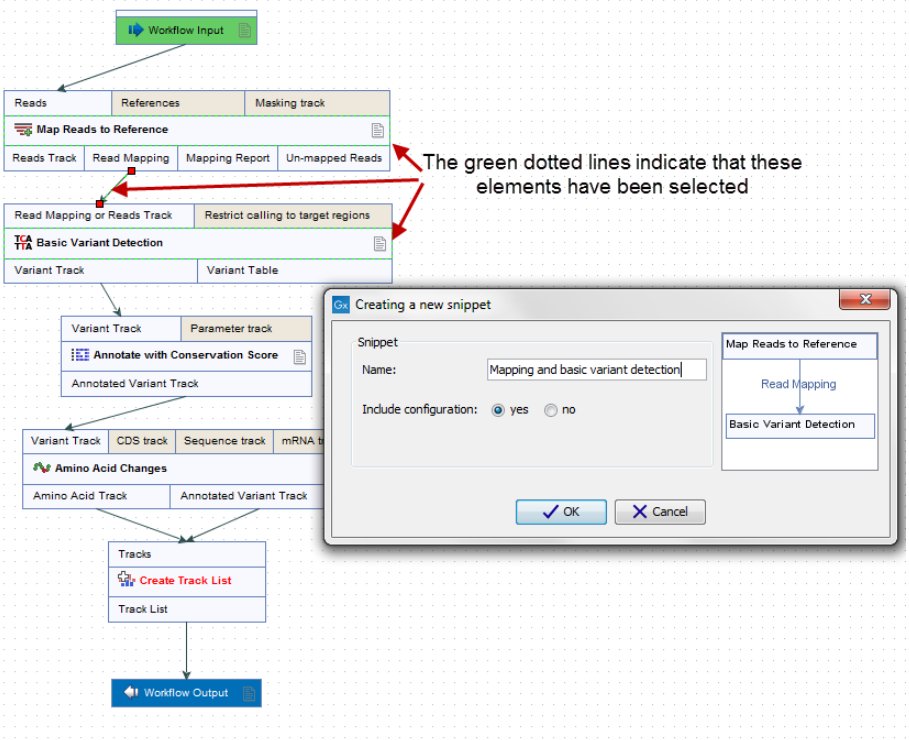


Figure 9.30: *In the "Create a new snippet" dialog you can name the snippet and select whether or not you would like to include the configuration. In the right-hand side of the dialog you can see the elements that are included in the snippet.*

- **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 elements" option that is shown in figure 9.34.

9.1.13 Change the order of tracks in the Genome Browser View

When modifying an existing workflow or creating a custom workflow that include the tool "**Create New Genome Browser View**" you may want to be able to adjust the order in which the tracks are shown in the Genome Browser View. To do this, display a view of the workflow layout, click once on the top part of the tool "Create New Genome Browser View" labeled "Tracks" followed by a right-click. In the pop up menu that appears (figure 9.35), choose the option "Order Workflow Inputs".

This opens a new pop up window (figure 9.36 where you can see a list of all the inputs that are connected with the input channel of the "Create New Genome Browser View" tool. Use the arrows found in the left-hand side to move the tracks up or down until you have the desired track

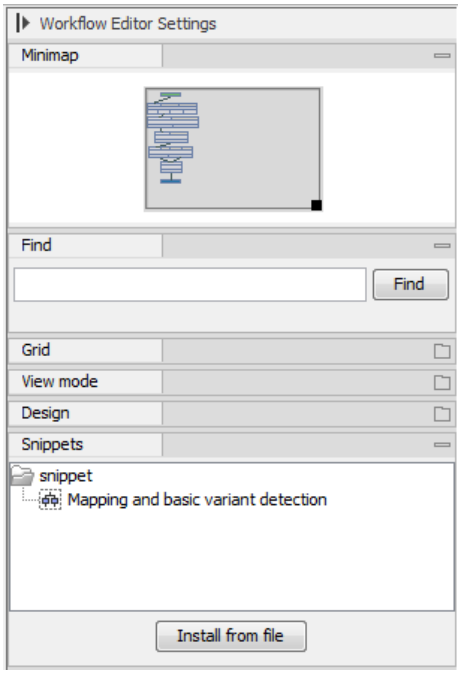


Figure 9.31: *When a snippet is installed, it appears in the Side Panel under the "Snippets" tab.*

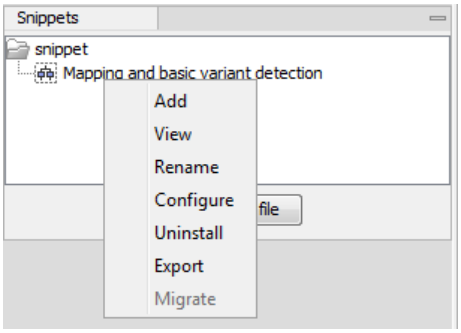


Figure 9.32: *Right-clicking on an installed snippet brings up a range of different options.*

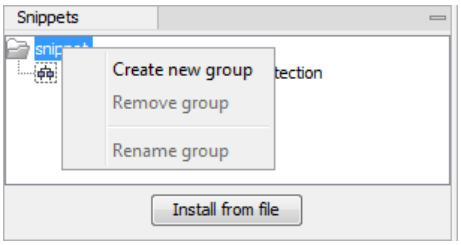


Figure 9.33: *Right-clicking on the snippet top-level folder makes it possible to manipulate the groups.*

order in your Genome Browser View.

If the workflow also generated several variant tracks, the variant table generated from the uppermost read mapping will open in split view with the Genome Browser View. By changing the Order of Inputs you can thus also influence which variant table should open in split view.

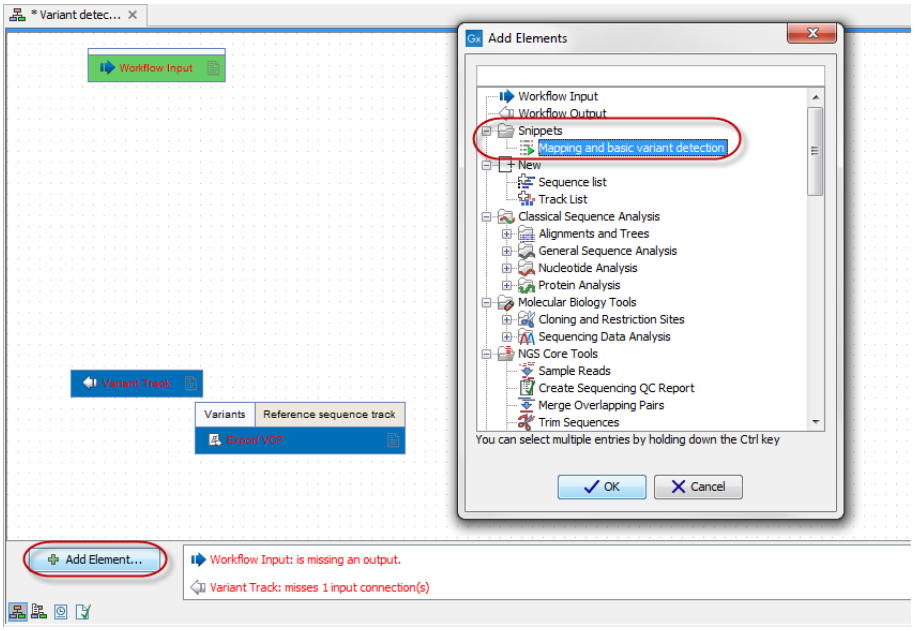


Figure 9.34: *Snippets can be added to a workflow in the workflow editor using the 'Add Elements' button found in the lower left corner.*

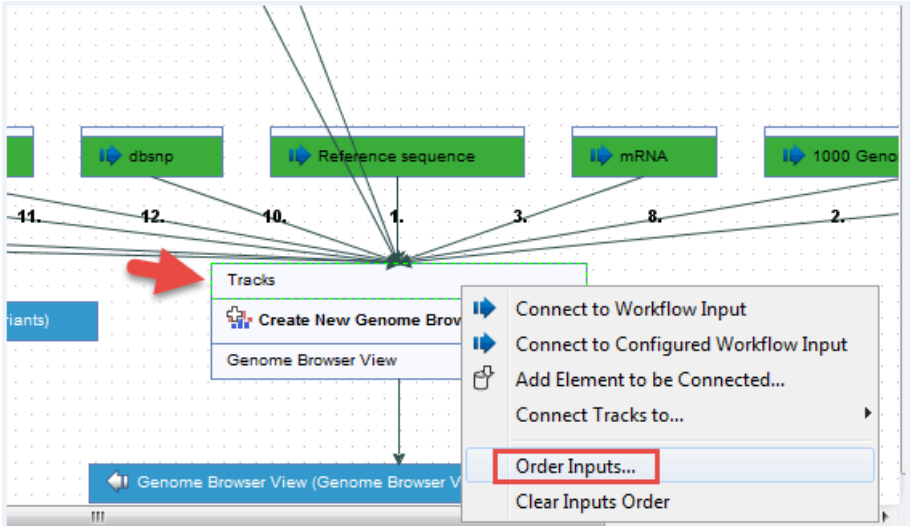


Figure 9.35: *Right click on the workflow layout and choose the option "Order Inputs...".*

9.2 Distributing and installing workflows

Once the workflow has been configured, you can use the **Run** button (see section 9.1.9) to process data through the workflow, but the real power of the workflow is its ability to be distributed and installed in the **Toolbox** alongside the other tools that come with the *Biomedical Genomics Workbench*, as well as the ability to install the same workflow on a *CLC Genomics Server*. The mechanism for distributing the workflow is a workflow installer file which can be created from the workflow editor and distributed and installed in any Workbench or Server.

9.2.1 Creating a workflow installation file

At the bottom of the workflow editor, click the **Create Installer** button (or use the shortcut Shift + Alt + I) to bring up a dialog where you provide information about the workflow to be distributed

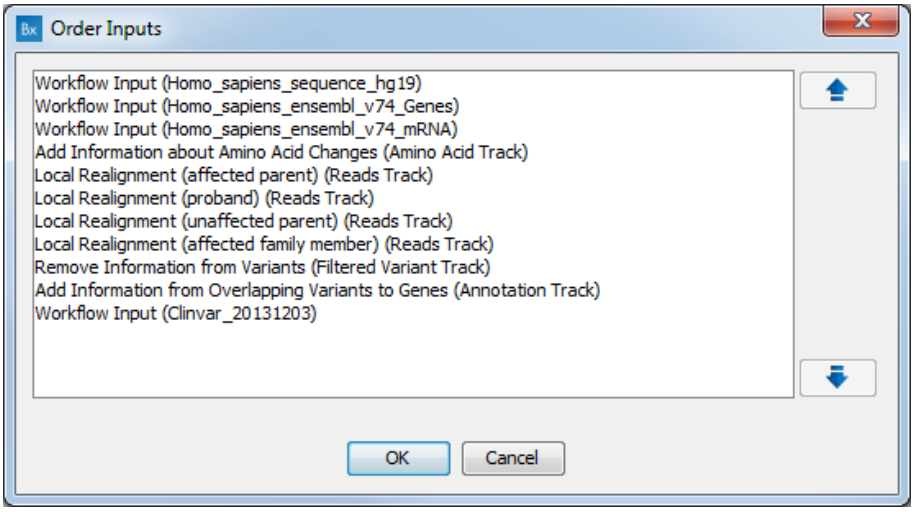


Figure 9.36: *Example of Order Inputs window that appears when choosing the option "Order Inputs...".*

(see an example in figure 9.37).

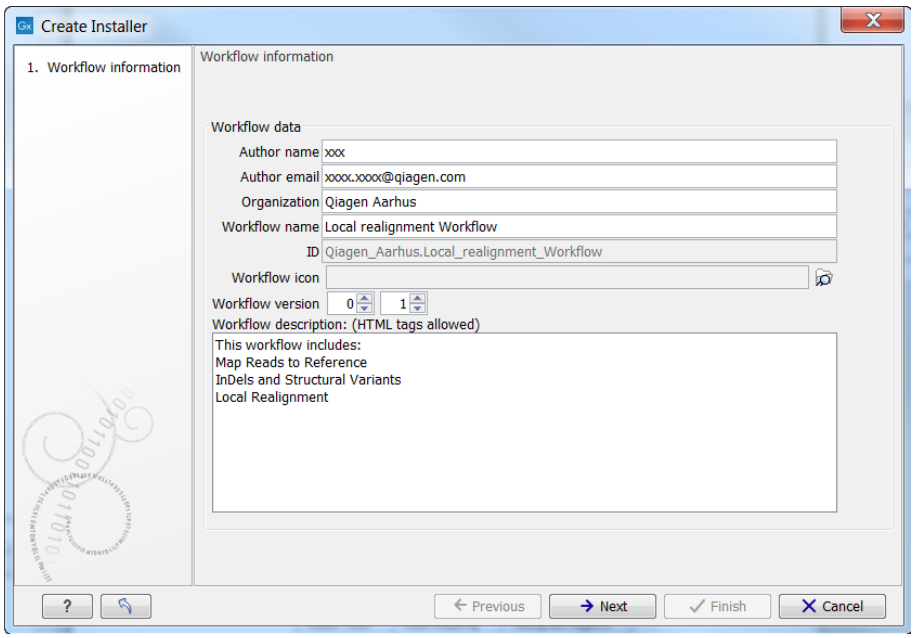


Figure 9.37: *Workflow information for the installer.*

Author information Providing name, email and organization of the author of the workflow. This will be visible for users installing the workflow and will enable them to look up the source of the workflow any time. The organization name is important because it is part of the workflow id (see more in section 9.2.4)

Workflow name The workflow name is based on the name used when saving the workflow in the **Navigation Area**. The workflow name is essential because it is used as part of the workflow id (see more in section 9.2.4). The workflow name can be changed during the installation of the workflow. This is useful whenever you have a workflow that you would like to use e.g. with small variations. The original workflow name will remain the same in the **Navigation**

Area - only the installed workflow will receive the customized name.

ID The final id of the workflow.

Workflow icon An icon can be provided. This will show up in the installation overview and in the **Toolbox** once the workflow is installed. The icon should be a 16 x 16 pixels gif or png file. If the icon is larger, it will automatically be resized to fit 16 x 16 pixels.

Workflow version A major and minor version can be provided.

Include original workflow file This will include the design file to be included with the installer. Once the workflow is installed in a workbench, you can extract the original workflow file and modify it.

Workflow description Provide a textual description of the workflow. This information will be displayed when a user mouses-over the name of the installed Workflow in the Workbench Toolbox, and is also presented in the Description tab for that Workflow in the Manage Workflows tool, described in section 9.2.3. Simple HTML tags are allowed (should be HTML 3.1 compatible, see `http://www.w3.org/TR/REC-html32`).

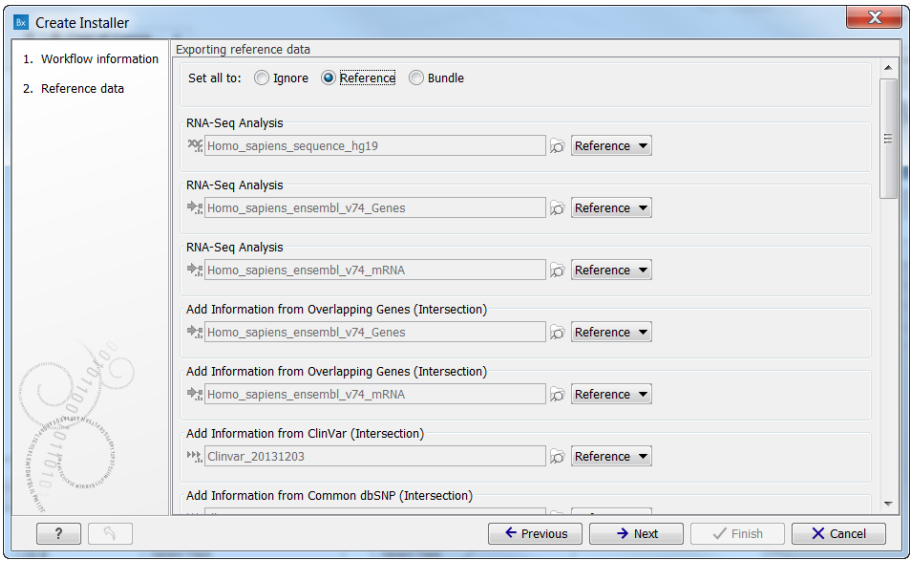


Figure 9.38: *Bundling data with the workflow installer.*

If you configured any of the workflow elements with data, clicking **Next** will give you the following options (see figure 9.38):

- **Ignore** This will exclude these reference data from the workflow.
- **Reference** This option can be used to include reference data from a shared CLC_References directory in a workflow without bundling the reference data with the workflow. Instead the reference data is included in the workflow by pointing at the shared CLC_References directory. This is particularly useful when working with large reference data.
- **Bundle** Includes data in the workflow by bundling the reference data with the workflow. **Note!** Bundling data should only be used to bundle small data sets with the workflow installer.

Click **Next** and you will be asked to specify where to install the workflow (figure 9.39). You can install your workflow directly on your local computer. If you are logged on a server and are the administrator, the option "Install the workflow on the current server" will be enabled. Finally, you can select to save the workflow as a .cpw file that can be installed on another computer. Click **Finish**. This will install the workflow directly on the selected destination. If you have selected to save the workflow for installation on another computer, you will be asked where to save the file after clicking **Finish**. If you chose to bundle data with your workflow installation, you will be asked for a location to put the bundled data on the workbench. Installing a workflow with bundled data on a server, the data will be put in a folder created in the first writable persistence location. Should this location not suit your needs, you can always move it afterwards, using the normal persistence operations.

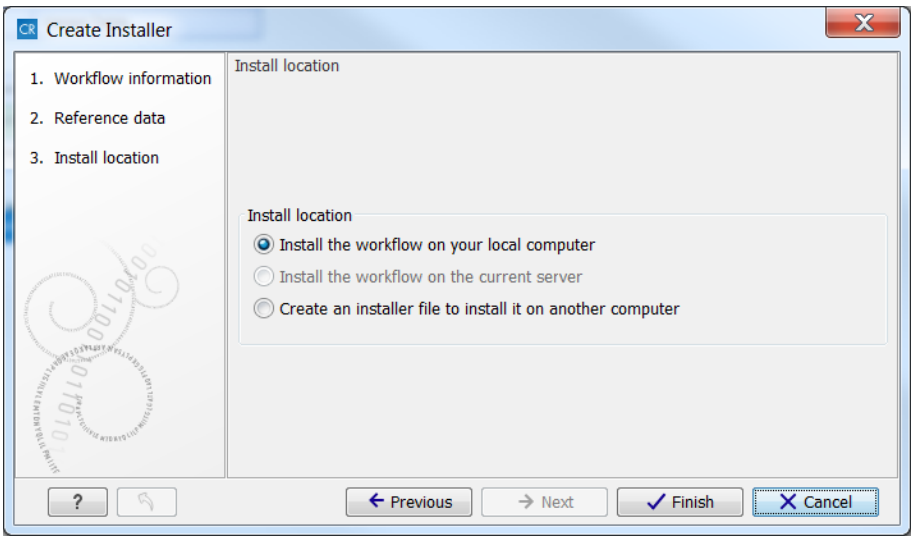


Figure 9.39: *Select whether the workflow should be installed on your local computer or on the current server. A third option is to create an installer file (.cpw) that can be installed on another computer.*

In cases where an existing workflow that has already been installed is modified, the workflow must be reinstalled. This can be done by first saving the workflow after it has been modified and then pressing the **Create Installer** button. Click through the wizard and select whether you wish to install the modified workflow on your local computer or on a server. Press **Finish**. This will open a pop-up dialog "Workflow is already installed" (figure 9.40) with the option that you can force the installation. This will uninstall the existing workflow and install the modified version of the workflow. **Note!** When forcing installation of the modified workflow, the configuration of the original workflow will be lost.

9.2.2 Installing a workflow

Workflow .cpw files can be installed on a Workbench using the workflow manager:

Help | Manage Workflows ( )

or press the "Workflows" button ( ) in the toolbar and then select "Manage Workflow..." ( ).

To install a workflow, click on Install from File and select a .cpw file. If the workflow has bundled data, you will be prompted for a location for that data. Once installed, the workflow will appear under the Custom Workflows tab (figure 9.41).

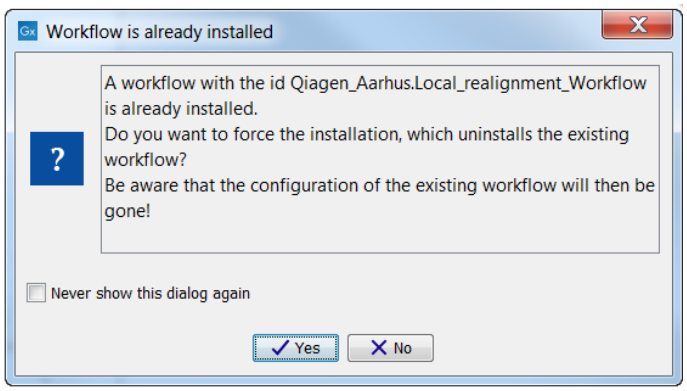


Figure 9.40: *Select whether you wish to force the installation of the workflow or keep the original workflow.*

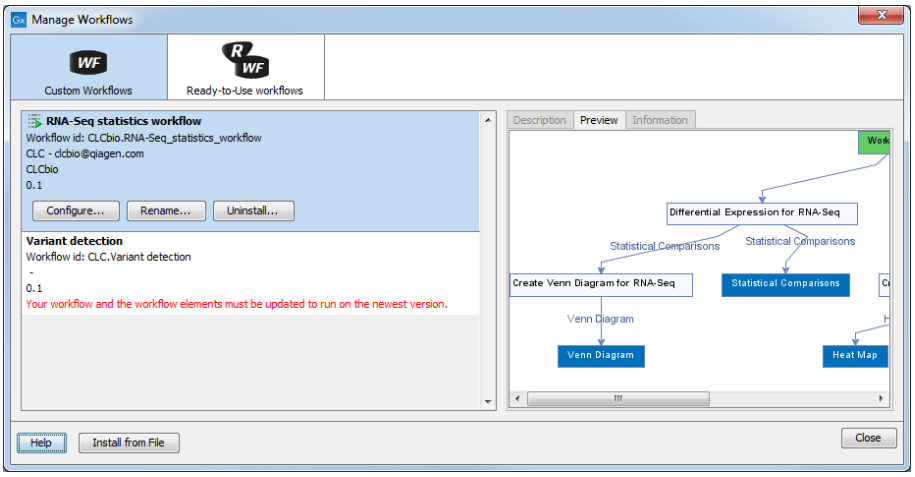


Figure 9.41: *Workflows available in the workflow manager. Note the alert on the "Variant detection" workflow, that means that this workflow needs to be updated.*

Information on how to create a .cpw file can be found section 9.2.1.

9.2.3 Managing workflows

Workflows can be managed from the workflow manager:

Help | Manage Workflows ( )

or using the "Workflows" button ( ) in the toolbar and then select "Manage Workflow..." ( ).

The workflow manager lists Custom workflows and Ready-to-Use workflows, but the functionalities described below (Configure, Rename, and Uninstall) are only available to custom workflows. You can always create a copy of a Ready-to-Use workflow (by opening the Ready-to-Use workflow and saving a copy in your Navigation Area) to enable the options described below.

Configure Select the workflow of interest and click on the button labeled Configure. You will be presented with a dialog listing all the reference data that need to be selected. An example is shown in figure 9.42.

This dialog also allows you to lock parameters of the workflow (see more about locking in section 9.1.3).

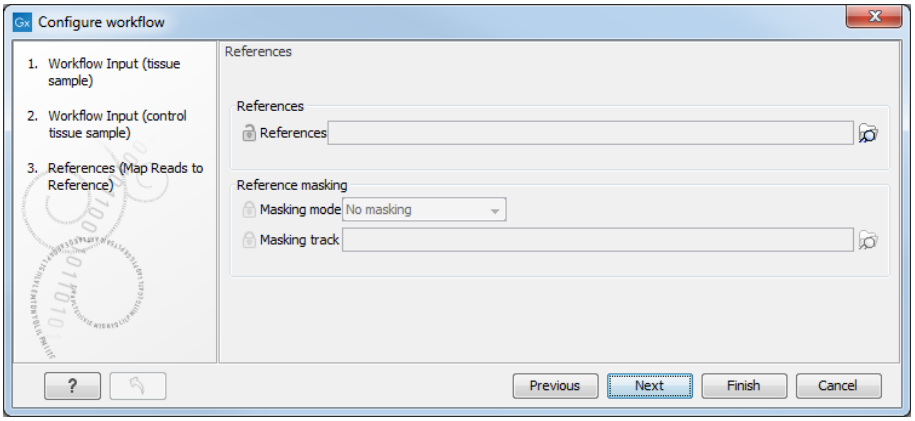


Figure 9.42: *Configuring parameters for the workflow.*

Note that if the workflow is intended to be executed on a server, it is important to select reference data that is located on the server.

Rename In addition to the configuration option, it is also possible to rename the workflow. This will change the name of the workflow in the **Toolbox**. The workflow id (see below) remains the same. To rename an element right click on the element name in the Navigation Area and select "Rename" or click on the F2 button.

Uninstall Use this button to install a workflow.

Description, Preview and Information In the right side of the window, you will find three tabs. **Description** contains the description that was entered when creating the workflow installer (see figure 9.37), the **Preview** shows a graphical representation of the workflow (figure 9.43), and finally you can get **Information** about the workflow (figure 9.44).

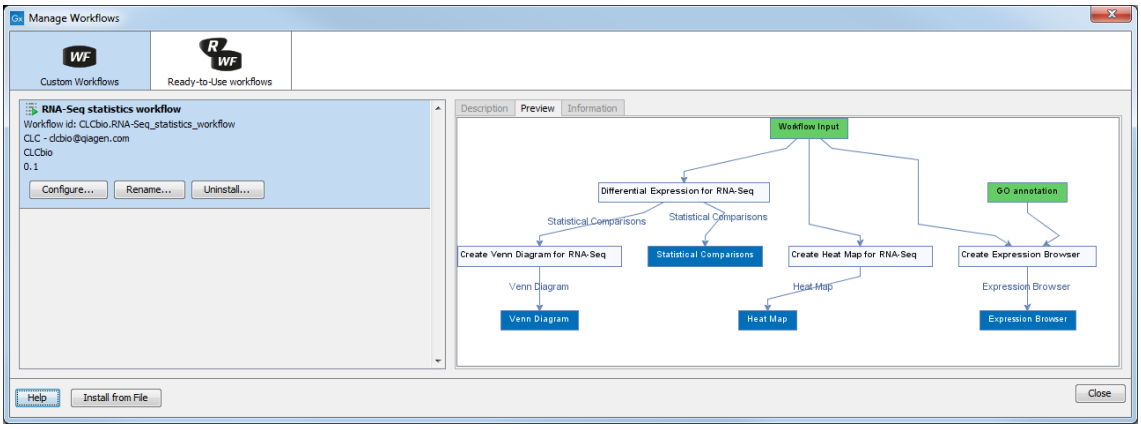


Figure 9.43: *Preview of the workflow.*

The "Information" field (figure 9.44) contains the following:

Build id The date (day month year) followed by the time (based on a 24 hour time) when the workflow was exported to a file through the Installation button at the bottom of the workflow window. If the workflow was installed locally without going through a file, the build ID will reflect the time of installation.

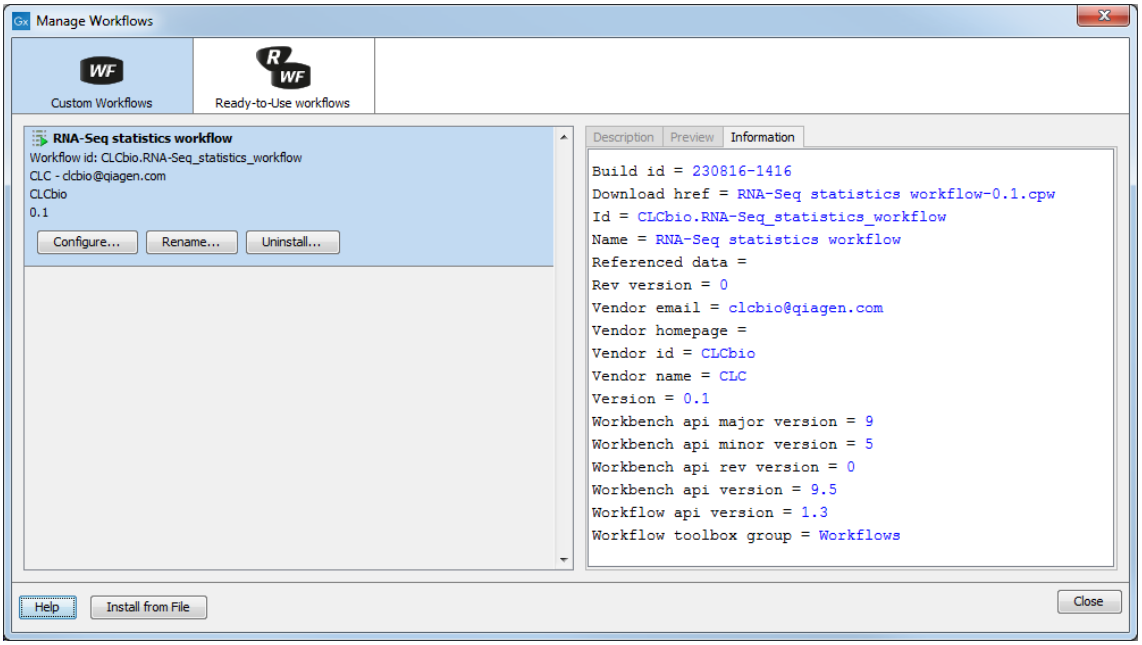


Figure 9.44: *Workflow identification and versioning.*

Download href The name of the workflow .cpw file

Id The unique id of a workflow, by which the workflow is identified

Major version The major version of the workflow

Minor version The minor version of the workflow

Name Name of workflow

Rev version Revision version. The functionality is activated but currently not in use

Vendor id ID of vendor that has created the workflow

Version <Major version>.<Minor version>

Workbench api version Workbench version

Workflow api version Workflow version (a technical number that can be used for troubleshooting)

9.2.4 Workflow identification and versioning

A workflow has a version. The version is used to make it easy to distribute an improved version of the same workflow. To do this, create a new installer with an incremented version number. In order to install a new and updated version, the old one has to be uninstalled.

The way *Biomedical Genomics Workbench* checks whether a workflow already exists in a previous version is by looking at the workflow id. The id is a combination of the organization name and the name of the workflow itself as it is shown in the dialog shown in figure 9.37. Once installed this information is also available in the workflow manager (figure 9.44).

If you create two different workflows with the same name and using the same organization name when creating the installer, they cannot both be installed.

9.2.5 Automatic update of workflow elements

Tools included in a workflow are versioned. They will initially be the same version as is present in the software being used to design the workflow. If one or more tools included in a workflow are updated, through upgrading a Workbench or plugins, the workflow must also be updated. The need to update a workflow is indicated by the message "Your workflow and the workflow elements must be updated to run on the newest version.", which will be shown:

- when you launch a workflow that needs to be updated.
- in the information on the left hand side of the "Manage Workflows..." ( ) tool for any workflows that need to be updated (see figure 9.45). This tool can be launched from the menu under the "Workflows" button ( ) in the top toolbar of the Workbench.

If a workflow needs to be updated, it must be updated before it can be used.

Please note that:

- When you update a workflow, the older version is overwritten.
- If new parameters have been added to a tool as part of the update, these parameters will be set to their default values within the updated workflow.

Updating installed workflows

To update an installed workflow:

- Open the "Manage Workflows..." ( ) tool by selecting it after clicking on the "Workflows" button ( ) in the top toolbar of the Workbench.
- Select the workflow that needs to be updated. Workflows you installed directly will be under the "Custom Workflows" tab. Other workflows will be listed under the "Ready-to-use workflows" tab.
- Click on the "Update" button.

To update a workflow you must have permission to write to the area the workflow is stored in. For workflows you installed directly, you will normally be able to do this when running the Workbench as you usually do. To update workflows distributed via plugins, it will usually mean running the Workbench as an administrative user.

Updating workflows in the Navigation Area

If a workflow stored in a CLC Data Location needs to be updated, this will become apparent when it is opened from the Navigation Area of the Workbench. In this case, an editor appears that lists the tools that need to be updated. See figure 9.46. The workflow must be updated before it will be opened in the Workflow editor, and edited or launched.

Click on the "OK" button at the bottom of the editor to update the workflow.

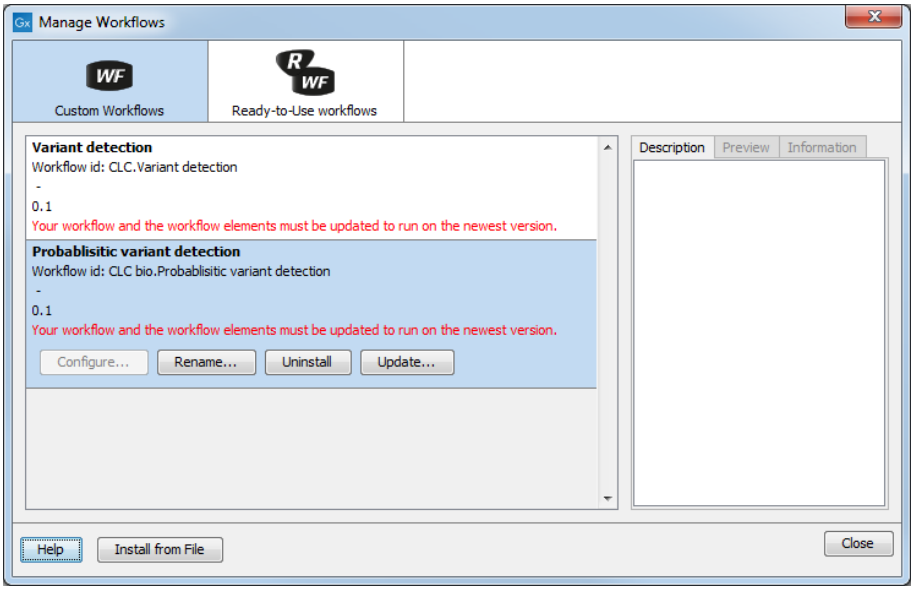


Figure 9.45: *A message is shown indicating that a workflow needs to be updated. Clicking on that workflow selects it, and a button labeled "Update" will be visible.*

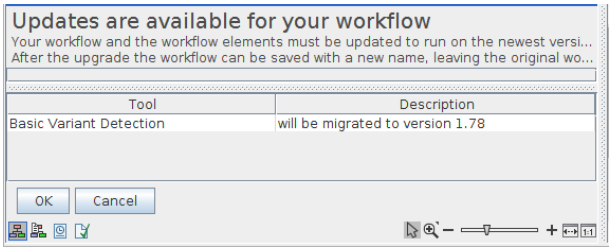


Figure 9.46: *When updates are available, an editor appears with information about which tools should be updated. Press "OK" to update the workflow.*

9.3 Executing a workflow

Once installed and configured, a workflow will appear in the **Toolbox** under **Workflows** ( ). If an icon was provided with the workflow installer this will also be shown (see figure 9.47).

The workflow is executed just as any other tool in the **Toolbox** by double-clicking or selecting it in the menu (or with the shortcut Ctrl + Enter). This will open a dialog where you provide input data and with options to run the workflow in batch mode (see section 8.3). In the last page of the dialog, you can preview all the parameters of the workflow, as well as the input data, before clicking "Next" to choose where to save the output, and then "Finish" to execute the workflow.

If you are connected to a *CLC Genomics Server*, you will be presented with the option to run the workflow locally on the Workbench or on the Server. When you are selecting where to run the workflow, you should also see a message should there be any missing configurations. There are more details about running Workflows here: `http://resources.qiagenbioinformatics.com/manuals/clcgenomicsserver/current/admin/index.php?manual=Workflows.html`.

When the workflow is started, you can see the log file with detailed information from each step in the process.

If the workflow is not properly configured, you will see that in the dialog when the workflow is

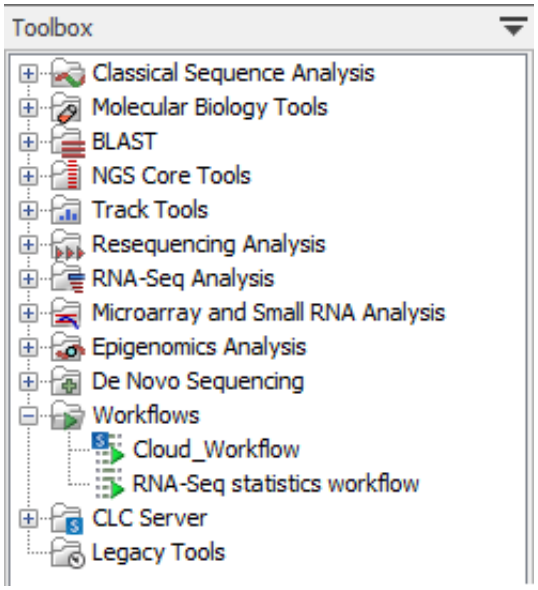


Figure 9.47: *A workflow is installed and ready to be used.*

started [1].

9.4 Open copy of ready-to-use workflow

In some situations it may be relevant to make adjustments to one of the existing pre-installed Ready-to-Use Workflows in the **Toolbox**. To do this you must first create a copy of the ready-to-use workflow that should be changed. A copy of a ready-to-use workflow found in the **Toolbox** under **Ready-to-Use Workflows** can be opened in the View Area by clicking once and then right-clicking on the name of the ready-to-use workflow and then selecting "Open Copy of Workflow" (figure 9.48).

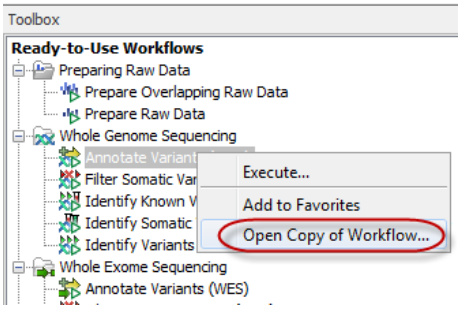


Figure 9.48: *A copy of a ready-to-use workflow can be opened from the Toolbox. The copied workflow will open in the View Area.*

This option makes it easy to view and modify the workflow. When the workflow has been modified, you must first save it at a destination of your choice in the **Navigation Area** as it is not possible to overwrite an original ready-to-use workflow. You can now choose to either run the workflow or to install the workflow in the **Toolbox** under **Workflows** ( ) (see section 9.2.1).

A copy of an installed and configured workflow found in the **Toolbox** under **Workflows** ( ) can

[1] If the workflow uses a tool that is part of a plugin, a missing plugin can also be the reason why the workflow is not enabled. A workflow can also become outdated because the underlying tools have changed since the workflow was created (see section 9.2.4)

be opened in the View Area in the exact same way as described for the ready-to-use workflows (see figure 9.49).

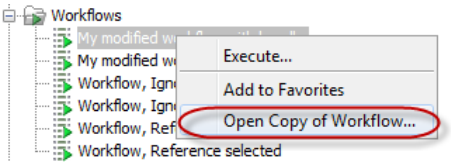


Figure 9.49: *A copy of an installed workflow can be opened from the Toolbox. The copied workflow will open in the View Area.*

An example of a copy of a workflow that has been opened in the **View Area** is shown in figure 9.50. At the bottom of the **View Area** the red text tells you that you must save the workflow before being able to install the workflow. When the workflow has been saved the button labeled "Installation" found in the lower right corner of the **View Area** will be enabled.

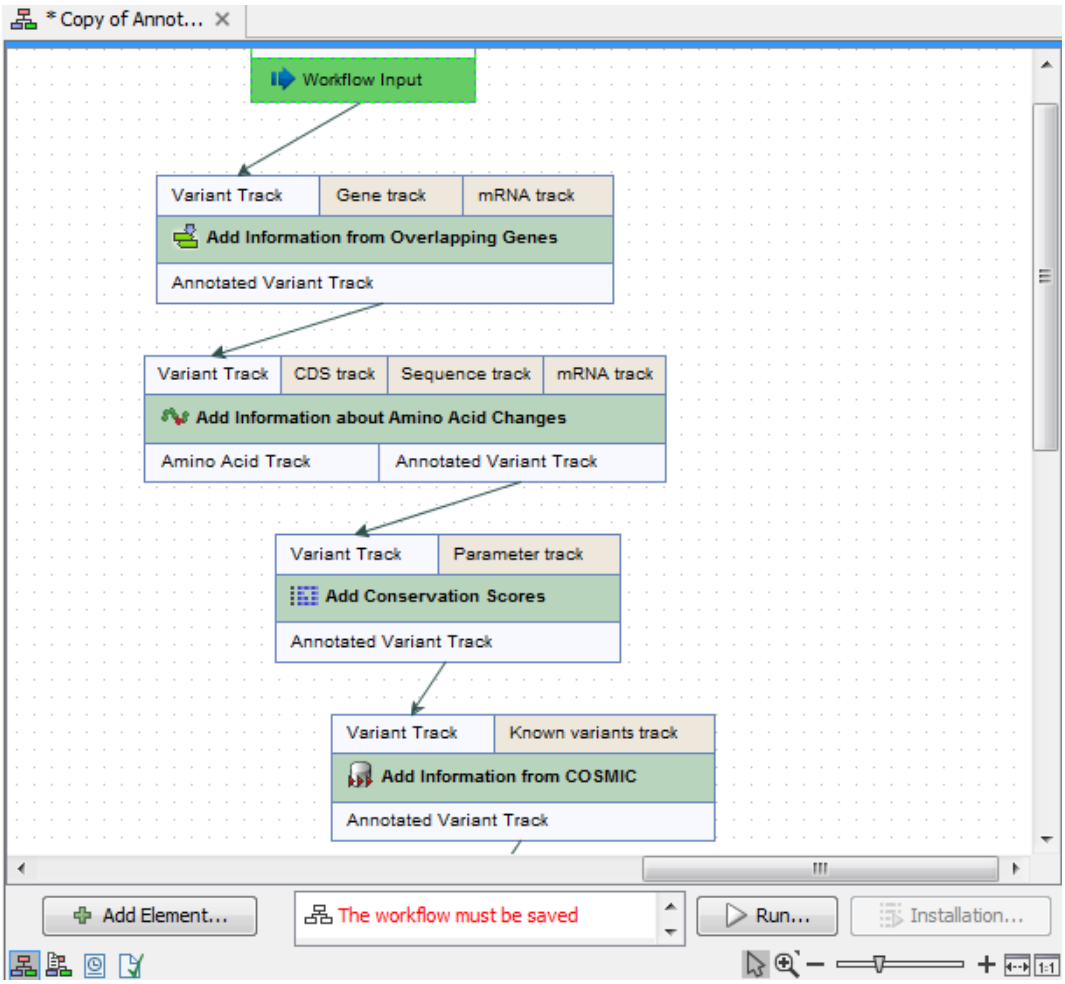


Figure 9.50: *A copy of a ready-to-use workflow after it has been opened in the View Area.*

Chapter 10

Viewing and editing sequences

Contents

Biomedical Genomics Workbench offers five different ways of viewing and editing single sequences as described in the first five sections of this chapter. Furthermore, this chapter also explains how to create a new sequence and how to gather several sequences in a sequence list.

10.1 View sequence

When you double-click a sequence in the **Navigation Area**, the sequence will open automatically, and you will see the nucleotides or amino acids. The zoom options described in section 2.2 allow you to e.g. zoom out in order to see more of the sequence in one view. There are a number of options for viewing and editing the sequence which are all described in this section.

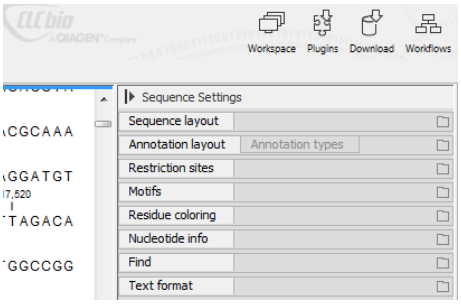


Figure 10.1: *Overview of the Side Panel which is always shown to the right of a view.*

10.1.1 Sequence settings in Side Panel

Each view of a sequence has a **Side Panel** located at the right side of the view (see figure 10.1.

When you make changes in the **Side Panel** the view of the sequence is instantly updated. To show or hide the **Side Panel**:

select the View | Ctrl + U

or **Click the ( ) at the top right corner of the Side Panel to hide | Click the ( ) to the right to show**

Below, each group of settings will be explained. Some of the preferences are not the same for nucleotide and protein sequences, but the differences will be explained for each group of settings.

Note! When you make changes to the settings in the **Side Panel**, they are not automatically saved when you save the sequence. Click **Save/restore Settings** ( ) to save the settings (see section 4.6 for more information).

Sequence Layout

These preferences determine the overall layout of the sequence:

- **Spacing.** Inserts a space at a specified interval:
  - **No spacing.** The sequence is shown with no spaces.
  - **Every 10 residues.** There is a space every 10 residues, starting from the beginning of the sequence.
  - **Every 3 residues, frame 1.** There is a space every 3 residues, corresponding to the reading frame starting at the first residue.
  - **Every 3 residues, frame 2.** There is a space every 3 residues, corresponding to the reading frame starting at the second residue.
  - **Every 3 residues, frame 3.** There is a space every 3 residues, corresponding to the reading frame starting at the third residue.
- **Wrap sequences.** Shows the sequence on more than one line.
  - **No wrap.** The sequence is displayed on one line.
  - **Auto wrap.** Wraps the sequence to fit the width of the view, not matter if it is zoomed in our out (displays minimum 10 nucleotides on each line).

  - **Fixed wrap.** Makes it possible to specify when the sequence should be wrapped. In the text field below, you can choose the number of residues to display on each line.

- **Double stranded.** Shows both strands of a sequence (only applies to DNA sequences).
- **Numbers on sequences.** Shows residue positions along the sequence. The starting point can be changed by setting the number in the field below. If you set it to e.g. 101, the first residue will have the position of -100. This can also be done by right-clicking an annotation and choosing **Set Numbers Relative to This Annotation**.
- **Numbers on plus strand.** Whether to set the numbers relative to the positive or the negative strand in a nucleotide sequence (only applies to DNA sequences).
- **Lock numbers.** When you scroll vertically, the position numbers remain visible. (Only possible when the sequence is not wrapped.)
- **Lock labels.** When you scroll horizontally, the label of the sequence remains visible.
- **Sequence label.** Defines the label to the left of the sequence.
  - Name (this is the default information to be shown).
  - Accession (sequences downloaded from databases like GenBank have an accession number).
  - Latin name.
  - Latin name (accession).
  - Common name.
  - Common name (accession).
- **Matching residues as dots** Residues in aligned sequences identical to residues in the first (reference) sequence will be presented as dots. An option that is only available for "Alignments" and "Read mappings".

Annotation Layout and Annotation Types See section 10.3.1.

Restriction sites

See section 10.1.1.

Motifs

In the **Side Panel** of sequence views, there is a group called **Motifs** (see figure 10.2).

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 10.3.

This case shows the CMV promoter primer sequence which is one of the pre-defined motifs in *Biomedical Genomics Workbench*. The motif is per default shown as a faded arrow with no text. The direction of the arrow indicates the strand of the motif.

Placing the mouse cursor on the arrow will display additional information about the motif as illustrated in figure 10.4.

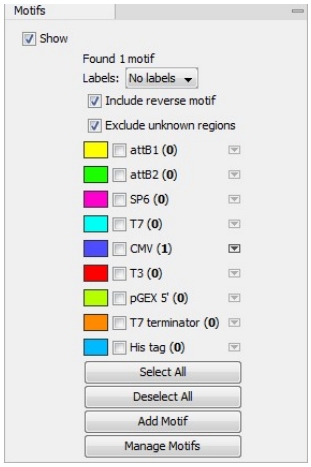


Figure 10.2: *Dynamic motifs in the Side Panel.*

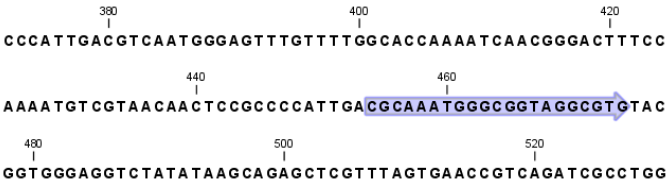


Figure 10.3: *Showing dynamic motifs on the sequence.*

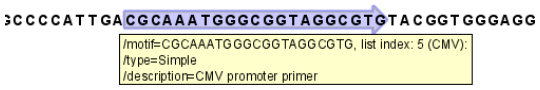


Figure 10.4: *Showing dynamic motifs on the sequence.*

To add **Labels** to the motif, select the **Flag** or **Stacked** option. They will put the name of the motif as a flag above the sequence. The stacked option will stack the labels when there is more than one motif so that all labels are shown.

Residue coloring

These preferences make it possible to color both the residue letter and set a background color for the residue.

- **Non-standard residues.** For nucleotide sequences this will color the residues that are not C, G, A, T or U. For amino acids only B, Z, and X are colored as non-standard residues.
  - **Foreground color.** Sets the color of the letter. Click the color box to change the color.
  - **Background color.** Sets the background color of the residues. Click the color box to change the color.
- **Rasmol colors.** Colors the residues according to the Rasmol color scheme.
  See `http://www.openrasmol.org/doc/rasmol.html`
  - **Foreground color.** Sets the color of the letter. Click the color box to change the color.
  - **Background color.** Sets the background color of the residues. Click the color box to change the color.

- **Polarity colors (only protein).** Colors the residues according to the following categories:
  - **Green** neutral, polar
  - **Black** neutral, nonpolar
  - **Red** acidic, polar
  - **Blue** basic ,polar
  - As with other options, you can choose to set or change the coloring for either the residue letter or its background:
    * **Foreground color.** Sets the color of the letter. Click the color box to change the color.
    * **Background color.** Sets the background color of the residues. Click the color box to change the color.
- **Trace colors (only DNA).** Colors the residues according to the color conventions of chromatogram traces: A=green, C=blue, G=black, and T=red.
  - **Foreground color.** Sets the color of the letter.
  - **Background color.** Sets the background color of the residues.

Nucleotide info

These preferences only apply to nucleotide sequences.

- **Color space encoding.** Lets you define a few settings for how the colors should appear.

  Infer encoding This is used if you want to display the colors for non-color space sequence (e.g. a reference sequence). The colors are then simply inferred from the sequence.

  Show corrections This is only relevant for mapping results - it will show where the mapping process has detected color errors.

  Hide unaligned This option determines whether color for the unaligned ends of reads should be displayed. It also controls whether colors should be shown for gaps. The idea behind this is that these color dots will interfere with the color alignment, so it is possible to turn them off.

- **Translation.** Displays a translation into protein just below the nucleotide sequence. Depending on the zoom level, the amino acids are displayed with three letters or one letter. 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 10.1.2.
        - **+1 to -1.** Select one of the six reading frames.
        - **All forward/All reverse.** Shows either all forward or all reverse reading frames.
        - **All.** Select all reading frames at once. The translations will be displayed on top of each other.
    - **Table.** The translation table to use in the translation.
    - **Only AUG start codons.** For most genetic codes, a number of codons can be start codons (TTG, CTG, or ATG). These will be colored green, unless selecting the "Only AUG start codons" option, which will result in only the AUG codons colored in green.
    - **Single letter codes.** Choose to represent the amino acids with a single letter instead of three letters.

- **Trace data.** See section 33.1.

- **Quality scores.** For sequencing data containing quality scores, the quality score information can be displayed along the sequence.

    - **Show as probabilities.** Converts quality scores to error probabilities on a 0-1 scale, i.e. not log-transformed.
    - **Foreground color.** Colors the letter using a gradient, where the left side color is used for low quality and the right side color is used for high quality. The sliders just above the gradient color box can be dragged to highlight relevant levels. The colors can be changed by clicking the box. This will show a list of gradients to choose from.
    - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
    - **Graph.** The quality score is displayed on a graph (Learn how to export the data behind the graph in section 6.8).
        - **Height.** Specifies the height of the graph.
        - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
        - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

- **G/C content.** Calculates the G/C content of a part of the sequence and shows it as a gradient of colors or as a graph below the sequence.

    - **Window length.** Determines the length of the part of the sequence to calculate. A window length of 9 will calculate the G/C content for the nucleotide in question plus the 4 nucleotides to the left and the 4 nucleotides to the right. A narrow window will focus on small fluctuations in the G/C content level, whereas a wider window will show fluctuations between larger parts of the sequence.
    - **Foreground color.** Colors the letter using a gradient, where the left side color is used for low levels of G/C content and the right side color is used for high levels of G/C content. The sliders just above the gradient color box can be dragged to highlight relevant levels of G/C content. The colors can be changed by clicking the box. This will show a list of gradients to choose from.

  - **Background color.** Sets a background color of the residues using a gradient in the same way as described above.
  - **Graph.** The G/C content level is displayed on a graph (Learn how to export the data behind the graph in section 6.8).
    * **Height.** Specifies the height of the graph.
    * **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
    * **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. For Colors, the color box is replaced by a gradient color box as described under Foreground color.

- **Secondary structure.** Allows you to choose how to display a symbolic representation of the secondary structure along the sequence.

Protein info

These preferences only apply to proteins. The first nine items are different hydrophobicity scales.

- **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 an 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 http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Dynamic_restriction_sites.html.

10.1.2 Selecting parts of the sequence

You can select parts of a sequence:

Click Selection ( ) in Toolbar | Press and hold down the mouse button on the sequence where you want the selection to start | move the mouse to the end of the selection while holding the button | release the mouse button

Alternatively, you can search for a specific interval using the find function described above.

If you have made a selection and wish to adjust it:

drag the edge of the selection (you can see the mouse cursor change to a horizontal arrow

or **press and hold the Shift key while using the right and left arrow keys to adjust the right side of the selection.**

If you wish to select the entire sequence:

double-click the sequence name to the left

Selecting several parts at the same time (multiselect) You can select several parts of sequence by holding down the **Ctrl** button while making selections. Holding down the **Shift** button lets you extend or reduce an existing selection to the position you clicked.

To select a part of a sequence covered by an annotation:

right-click the annotation | **Select annotation**

or **double-click the annotation**

To select a fragment between two restriction sites that are shown on the sequence:

double-click the sequence between the two restriction sites

(Read more about restriction sites in section 10.1.1.)

Open a selection in a new view A selection can be opened in a new view and saved as a new sequence:

right-click the selection | **Open selection in New View (**  **)**

This opens the annotated part of the sequence in a new view. The new sequence can be saved by dragging the tab of the sequence view into the **Navigation Area**.

The process described above is also the way to manually translate coding parts of sequences (CDS) into protein. You simply translate the new sequence into protein. This is done by:

right-click the tab of the new sequence | **Toolbox** | **Nucleotide Analysis (****)**| **Translate to Protein (**  **)**

A selection can also be copied to the clipboard and pasted into another program:

make a selection | **Ctrl + C (⌘ + C on Mac)**

Note! The annotations covering the selection will not be copied.

A selection of a sequence can be edited as described in the following section.

10.1.3 Editing the sequence

When you make a selection, it can be edited by:

right-click the selection | **Edit Selection (**  **)**

A dialog appears displaying the sequence. You can add, remove or change the text and click **OK**. The original selected part of the sequence is now replaced by the sequence entered in the dialog. This dialog also allows you to paste text into the sequence using Ctrl + V (⌘ + V on Mac).

If you delete the text in the dialog and press **OK**, the selected text on the sequence will also be deleted. Another way to delete a part of the sequence is to:

right-click the selection | **Delete Selection (**  **)**

If you wish to correct only one residue, this is possible by simply making the selection cover only one residue and then type the new residue.

Note When editing annotated nucleotide sequences, the annotation content is not updated automatically (but its position is). Please refer to section 10.3.3 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.

10.1.4 Sequence region types

The various annotations on sequences cover parts of the sequence. Some cover an interval, some cover intervals with unknown endpoints, some cover more than one interval etc. In the following, all of these will be referred to as *regions*. Regions are generally illustrated by markings (often arrows) on the sequences. An arrow pointing to the right indicates that the corresponding region is located on the positive strand of the sequence. Figure 10.5 is an example of three regions with separate colors.

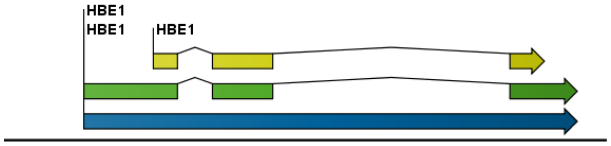


Figure 10.5: *Three regions on a human beta globin DNA sequence (HUMHBB).*

Figure 10.6 shows an artificial sequence with all the different kinds of regions.

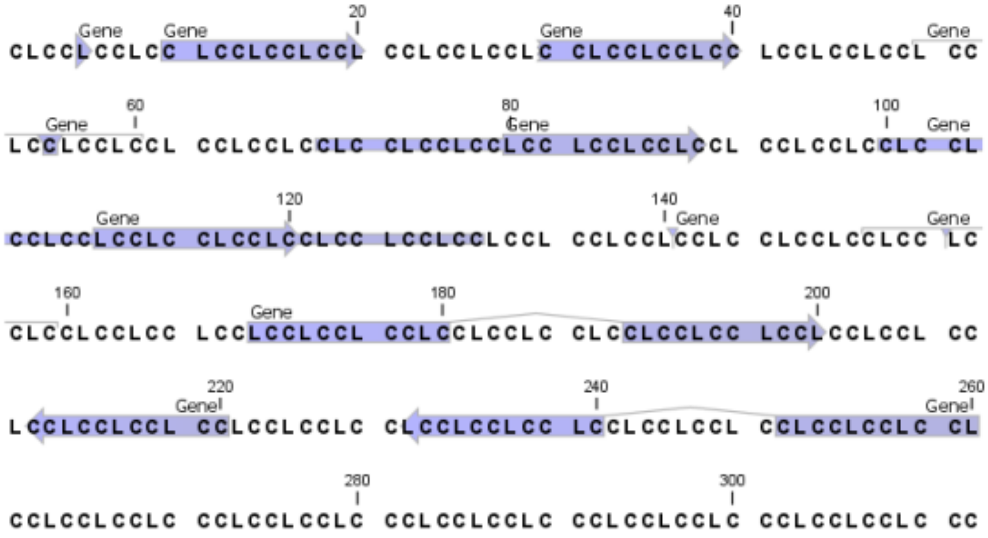


Figure 10.6: *Region #1: A single residue, Region #2: A range of residues including both endpoints, Region #3: A range of residues starting somewhere before 30 and continuing up to and including 40, Region #4: A single residue somewhere between 50 and 60 inclusive, Region #5: A range of residues beginning somewhere between 70 and 80 inclusive and ending at 90 inclusive, Region #6: A range of residues beginning somewhere between 100 and 110 inclusive and ending somewhere between 120 and 130 inclusive, Region #7: A site between residues 140 and 141, Region #8: A site between two residues somewhere between 150 and 160 inclusive, Region #9: A region that covers ranges from 170 to 180 inclusive and 190 to 200 inclusive, Region #10: A region on negative strand that covers ranges from 210 to 220 inclusive, Region #11: A region on negative strand that covers ranges from 230 to 240 inclusive and 250 to 260 inclusive.*

10.2 Circular DNA

A sequence can be shown as a circular molecule:

Select a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Select "Circular View" ( )

or **If the sequence is already open | Click "Show Circular View" ( ) at the lower left part of the view**

This will open a view of the molecule similar to the one in figure 10.7.

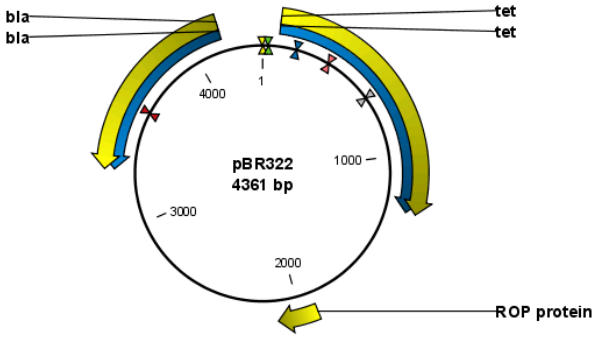


Figure 10.7: *A molecule shown in a circular view.*

This view of the sequence shares some of the properties of the linear view of sequences as described in section 10.1, but there are some differences. The similarities and differences are listed below:

- **Similarities**:
  - The editing options.
  - Options for adding, editing and removing annotations.
  - **Restriction Sites**, **Annotation Types**, **Find** and **Text Format** preferences groups.
- **Differences**:
  - In the **Sequence Layout** preferences, only the following options are available in the circular view: **Numbers on plus strand**, **Numbers on sequence** and **Sequence label**.
  - You cannot zoom in to see the residues in the circular molecule. If you wish to see these details, split the view with a linear view of the sequence
  - In the **Annotation Layout**, you also have the option of showing the labels as **Stacked**. This means that there are no overlapping labels and that all labels of both annotations and restriction sites are adjusted along the left and right edges of the view.

10.2.1 Using split views to see details of the circular molecule

In order to see the nucleotides of a circular molecule you can open a new view displaying a circular view of the molecule:

Press and hold the Ctrl button (⌘ on Mac) | click Show Sequence ( ) at the bottom of the view

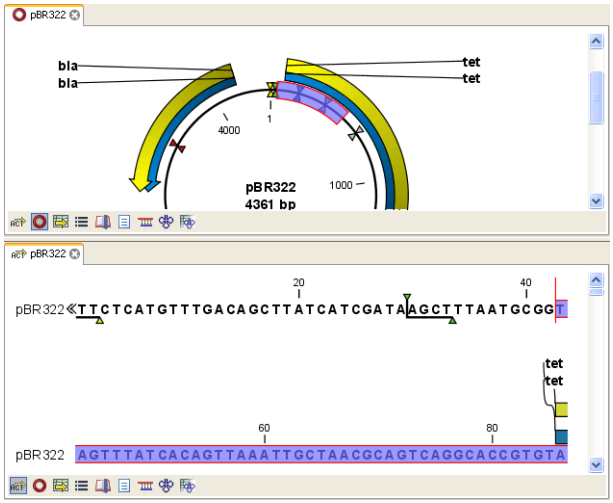


Figure 10.8: *Two views showing the same sequence. The bottom view is zoomed in.*

This will open a linear view of the sequence below the circular view. When you zoom in on the linear view you can see the residues as shown in figure 10.8.

Note! If you make a selection in one of the views, the other view will also make the corresponding selection, providing an easy way for you to focus on the same region in both views.

10.2.2 Mark molecule as circular and specify starting point

You can mark a DNA molecule as circular 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 10.9).

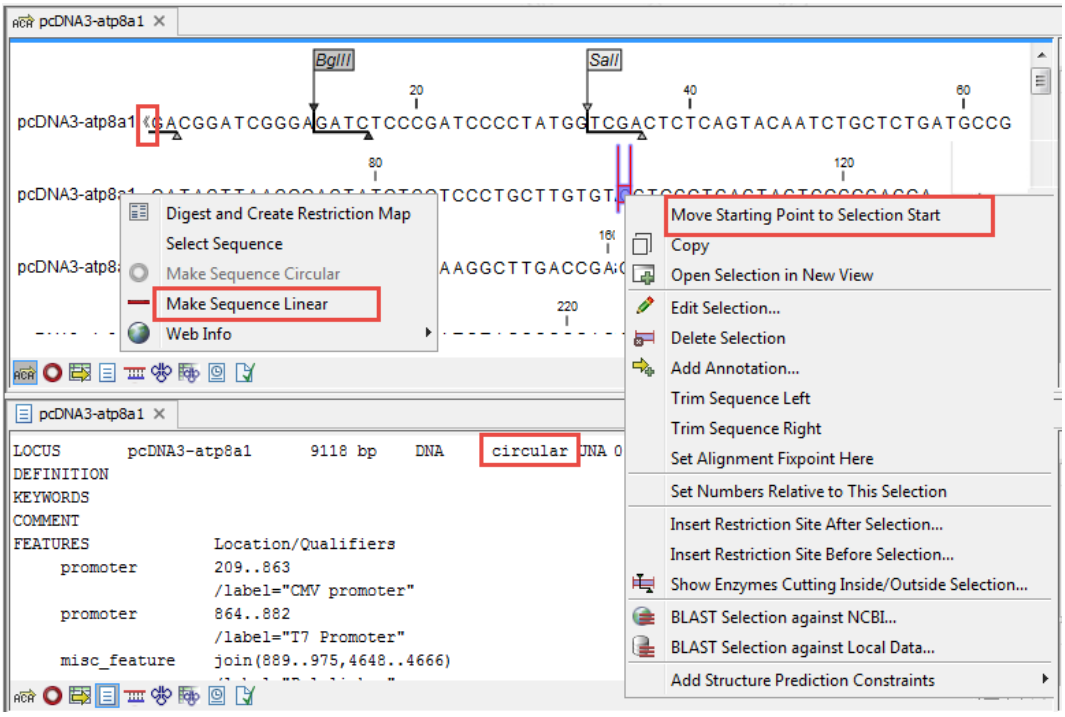


Figure 10.9: *Double angle brackets marks the start and end of a circular sequence seen in linear view. Below, the Text view of the same sequence shows the mention circular in the first line.*

In the Sequence view, a sequence marked as circular is indicated by the use of double angle brackets at the start and end of the sequence. The linear or circular status of a sequence can also be seen in the Locus line of the Text view for a Sequence, or in the Linear column of the Table view of a Sequence List.

The starting point of a circular sequence can be changed by selecting the position of the new starting point and right-clicking on that selection to choose the option **Move Starting Point to Selection Start** (figure 10.10).

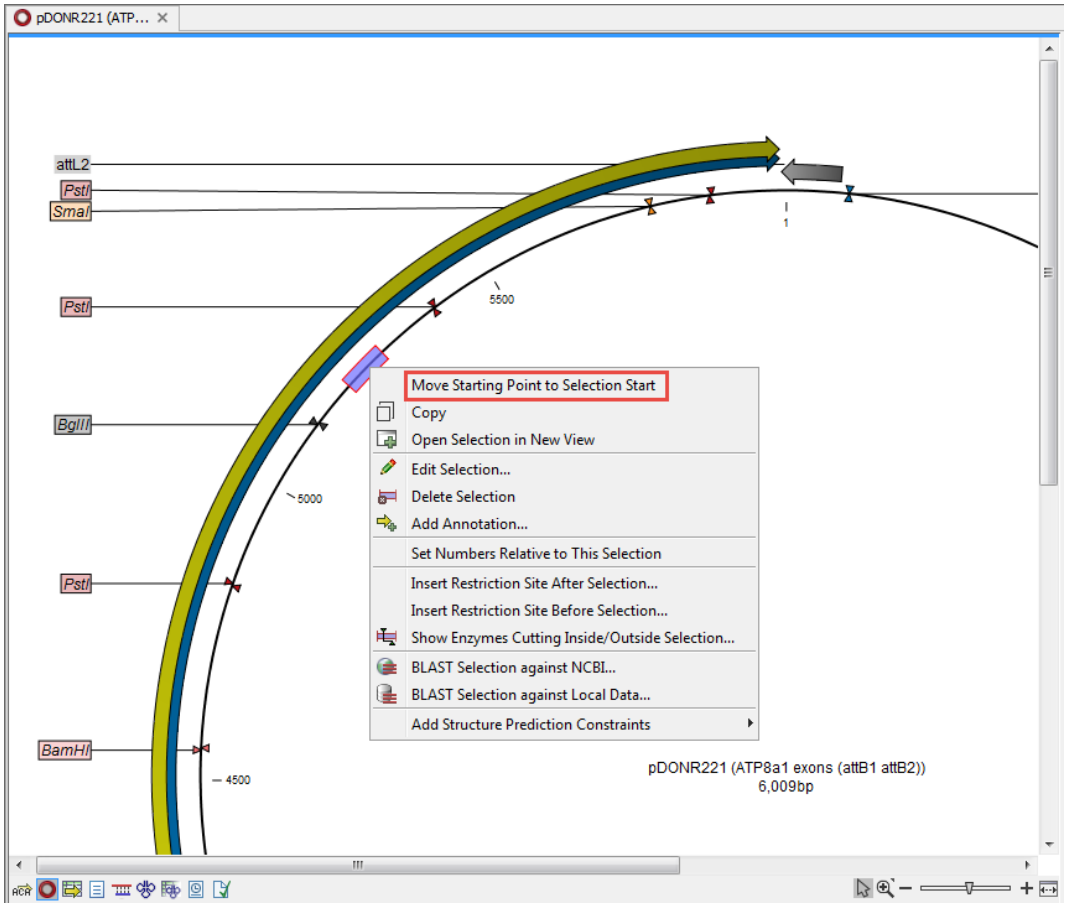


Figure 10.10: *Right-click on a circular sequence to move the starting point to the selected position.*

.

10.3 Working with annotations

Note! This section only applies to sequences that is not in track format e.g. sequences from Sanger sequencing.

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 *Biomedical Genomics Workbench*,

sequences can have annotations (GenBank, EMBL and Swiss-Prot format).

- The result of a number of analyses in *Biomedical Genomics 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 11.4.2), and atom groups defined on the structure transferred to sequence annotations or vica versa (section 11.4.3).
- You can manually add annotations to a sequence (described in the section 10.3.2).

Note! Annotations are included if you export the sequence in GenBank, Swiss-Prot, EMBL or CLC format. When exporting in other formats, annotations are not preserved in the exported file.

10.3.1 Viewing annotations

Annotations can be viewed in a number of different ways:

- As arrows or boxes in all views displaying sequences (sequence lists, alignments etc)
- In the table of annotations ( ).
- In the text view of sequences ( )

In the following sections, these view options will be described in more detail. In all the views except the text view ( ), annotations can be added, modified and deleted. This is described in the following sections.

View Annotations in sequence views

Figure 10.11 shows an annotation displayed on a sequence.

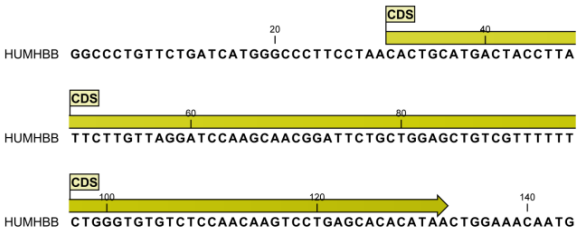


Figure 10.11: *An annotation showing a coding region on a genomic dna sequence.*

The various sequence views listed in section 10.3.1 have different default settings for showing annotations. However, they all have two groups in the **Side Panel** in common:

- **Annotation Layout**
- **Annotation Types**

The two groups are shown in figure 10.12.

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):

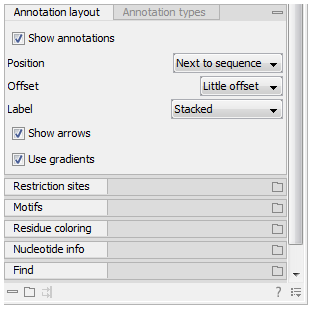


Figure 10.12: *The annotation layout in the Side Panel. The annotation types can be shown by clicking on the "Annotation types" tab.*

- **Show annotations.** Determines whether the annotations are shown.
- **Position.**
  - **On sequence.** The annotations are placed on the sequence. The residues are visible through the annotations (if you have zoomed in to 100%).
  - **Next to sequence.** The annotations are placed above the sequence.
  - **Separate layer.** The annotations are placed above the sequence and above restriction sites (only applicable for nucleotide sequences).
- **Offset.** If several annotations cover the same part of a sequence, they can be spread out.
  - **Piled.** The annotations are piled on top of each other. Only the one at front is visible.
  - **Little offset.** The annotations are piled on top of each other, but they have been offset a little.
  - **More offset.** Same as above, but with more spreading.
  - **Most offset.** The annotations are placed above each other with a little space between. This can take up a lot of space on the screen.
- **Label.** The name of the annotation can shown as a label. Additional information about the sequence is shown if you place the mouse cursor on the annotation and keep it still.
  - **No labels.** No labels are displayed.
  - **On annotation.** The labels are displayed in the annotation's box.
  - **Over annotation.** The labels are displayed above the annotations.
  - **Before annotation.** The labels are placed just to the left of the annotation.
  - **Flag.** The labels are displayed as flags at the beginning of the annotation.
  - **Stacked.** The labels are offset so that the text of all labels is visible. This means that there is varying distance between each sequence line to make room for the labels.
- **Show arrows.** Displays the end of the annotation as an arrow. This can be useful to see the orientation of the annotation (for DNA sequences). Annotations on the negative strand will have an arrow pointing to the left.
- **Use gradients.** Fills the boxes with gradient color.

In the **Annotation types** group, you can choose which kinds of annotations that should be displayed. This group lists all the types of annotations that are attached to the sequence(s) in the view. For sequences with many annotations, it can be easier to get an overview if you deselect the annotation types that are not relevant.

Unchecking the checkboxes in the **Annotation layout** will not remove this type of annotations them from the sequence - it will just hide them from the view.

Besides selecting which types of annotations that should be displayed, the **Annotation types** group is also used to change the color of the annotations on the sequence. Click the colored square next to the relevant annotation type to change the color.

This will display a dialog with 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 10.13).

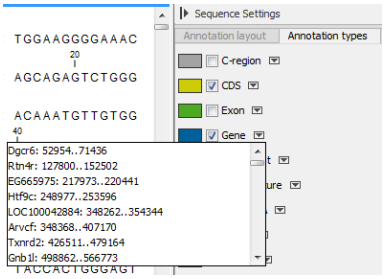


Figure 10.13: *Browsing the gene annotations on a sequence.*

Clicking an annotation in the list will select this region on the sequence. In this way, you can quickly find a specific annotation on a long sequence.

Note: A waved end on an annotation (figure 10.14) means that the annotation is torn, i.e., it extends beyond the sequence displayed. An annotation can be torn when a new, smaller sequence has been created from a larger sequence. A common example of this situation is when you select a section of a stand alone sequence and open it in a new view. If there are annotations present within this selected region that extend beyond the selection, then the selected sequence shown in the new view will exhibit these torn annotations.

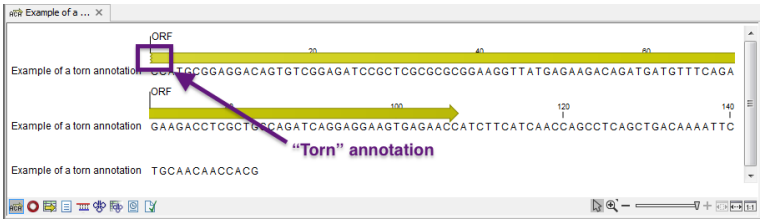


Figure 10.14: *Example of a torn annotation on a sequence.*

View Annotations in a table

Annotations can also be viewed in a table:

Select a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Annotation Table ( )

or **If the sequence is already open | Click Show Annotation Table ( ) at the lower left part of the view**

This will open a view similar to the one in figure 10.15).

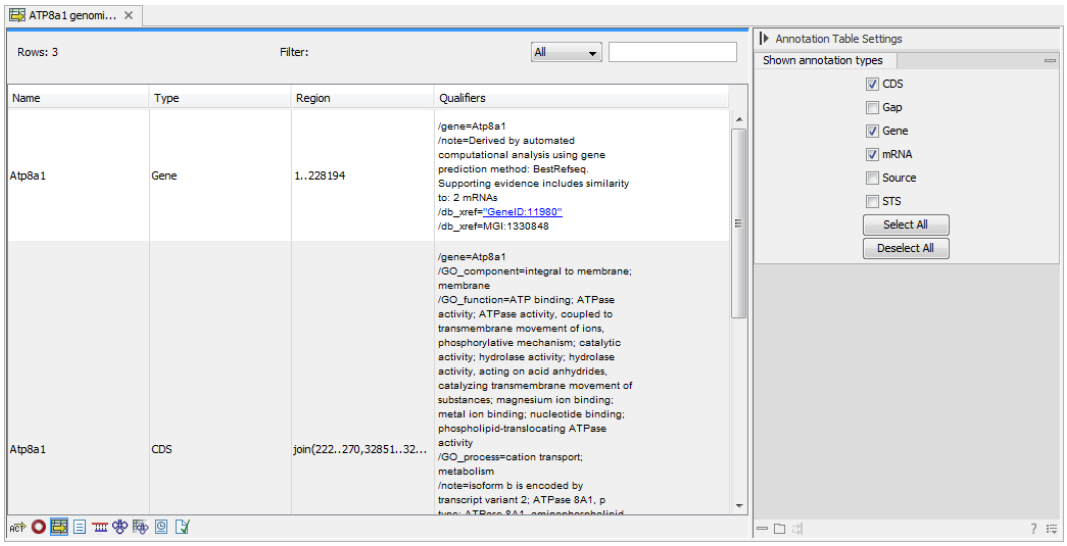


Figure 10.15: *A table showing annotations on the sequence.*

In the **Side Panel** you can show or hide individual annotation types in the table. E.g. if you only wish to see "gene" annotations, de-select the other annotation types so that only "gene" is selected.

Each row in the table is an annotation which is represented with the following information:

- **Name.**
- **Type.**
- **Region.**
- **Qualifiers.**

The Name, Type and Region for each annotation can be edited simply by double-clicking, typing the change directly, and pressing **Enter**.

This information corresponds to the information in the dialog when you edit and add annotations (see section 10.3.2).

You can benefit from this table in several ways:

- It provides an intelligible overview of all the annotations on the sequence.
- You can use the filter at the top to search the annotations. Type e.g. "UCP" into the filter and you will find all annotations which have "UCP" in either the name, the type, the region or the qualifiers. Combined with showing or hiding the annotation types in the **Side Panel**, this makes it easy to find annotations or a subset of annotations.

- You can copy and paste annotations, e.g. from one sequence to another.
- If you wish to edit many annotations consecutively, the double-click editing makes this very fast (see section 10.3.2).

10.3.2 Adding annotations

Adding annotations to a sequence can be done in two ways:

Open the sequence in a sequence view (double-click in the Navigation Area) | make a selection covering the part of the sequence you want to annotate[1] **| right-click the selection | Add Annotation ()**

or **Select a sequence in the Navigation Area and right-click on the file name | Hold the mouse over "Show" to enable a list of options | Annotation table () | right click anywhere in the annotation table | select Add Annotation ()**

This will display a dialog like the one in figure 10.16.

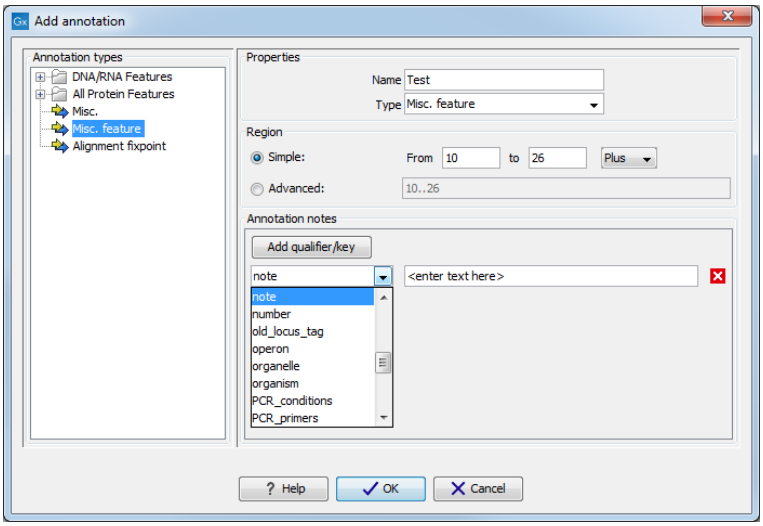


Figure 10.16: *The Add Annotation dialog.*

The left-hand part of the dialog lists a number of **Annotation types**. When you have selected an annotation type, it appears in **Type** to the right. You can also select an annotation directly in this list. Choosing an annotation type is mandatory. If you wish to use an annotation type which is not present in the list, simply enter this type into the **Type** field [2].

The right-hand part of the dialog contains the following text fields:

- **Name.** The name of the annotation which can be shown on the label in the sequence views. (Whether the name is actually shown depends on the **Annotation Layout** preferences, see section 10.3.1).
- **Type.** Reflects the left-hand part of the dialog as described above. You can also choose directly in this list or type your own annotation type.

[1] (See section 2.2.3 on how to make selections that are not contiguous.)

[2] Note that your own annotation types will be converted to "unsure" when exporting in GenBank format. As long as you use the sequence in CLC format, you own annotation type will be preserved

- **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 10.3.1) and in the yellow box which appears when you place the mouse cursor on the annotation. If you write a hyperlink in the **Key** text field, like e.g. "www.qiagenbioinformatics.com", it will be recognized as a hyperlink. Clicking the link in the annotation table will open a web browser.

Click **OK** to add the annotation.

Note! The annotation will be included if you export the sequence in GenBank, Swiss-Prot or CLC format. When exporting in other formats, annotations are not preserved in the exported file.

10.3.3 Edit annotations

To edit an existing annotation from within a sequence view:

> **right-click the annotation | Edit Annotation ( )**

This will show the same dialog as in figure 10.16, with the exception that some of the fields are filled out depending on how much information the annotation contains.

There is another way of quickly editing annotations which is particularly useful when you wish to edit several annotations.

To edit the information, simply double-click and you will be able to edit e.g. the name or the annotation type. If you wish to edit the qualifiers and double-click in this column, you will see the dialog for editing annotations.

Advanced editing of annotations

Sometimes you end up with annotations which do not have a meaningful name. In that case there is an advanced batch rename functionality:

> **Open the Annotation Table ( ) | select the annotations that you want to rename | right-click the selection | Advanced Rename**

This will bring up the dialog shown in figure 10.17.

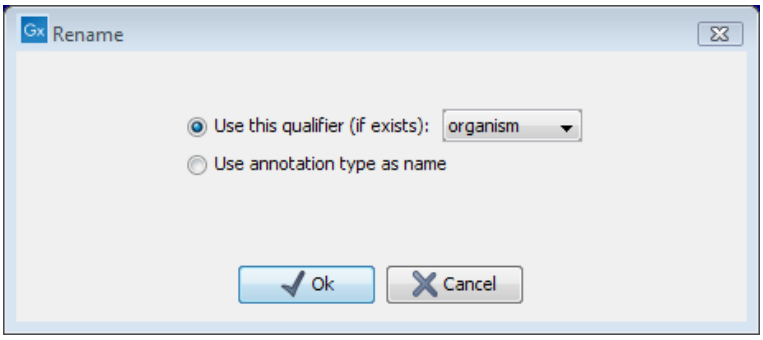


Figure 10.17: *The Advanced Rename dialog.*

In this dialog, you have two options:

- **Use this qualifier.** Use one of the qualifiers as name. A list of all qualifiers of all the selected annotations is shown. Note that if one of the annotations do not have the qualifier you have chosen, it will not be renamed. If an annotation has multiple qualifiers of the same type, the first is used for naming.
- **Use annotation type as name.** The annotation's type will be used as name (e.g. if you have an annotation of type "Promoter", it will get "Promoter" as its name by using this option).

A similar functionality for batch re-typing annotations is available in the right-click menu as well, in case your annotations are not typed correctly:

> **Open the Annotation Table ( ) | select the annotations that you want to retype | right-click the selection | Advanced Retype**

This will bring up the dialog shown in figure 10.18.

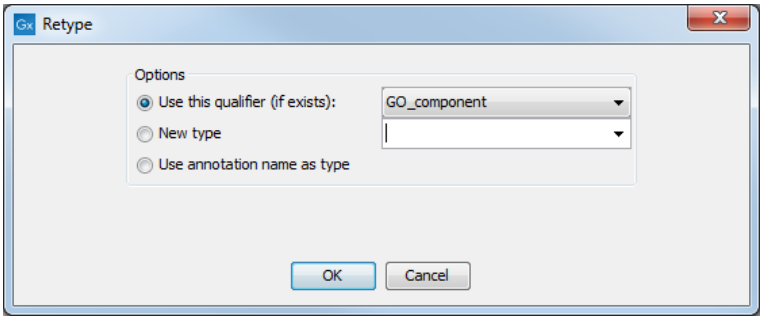


Figure 10.18: *The Advanced Retype dialog.*

In this dialog, you have two options:

- **Use this qualifier.** Use one of the qualifiers as type. A list of all qualifiers of all the selected annotations is shown. Note that if one of the annotations do not have the qualifier you have chosen, it will not be retyped. If an annotation has multiple qualifiers of the same type, the first is used for the new type.
- **New type**. You can select from a list of all the pre-defined types as well as enter your own annotation type. All the selected annotations will then get this type.
- **Use annotation name as type.** The annotation's name will be used as type (e.g. if you have an annotation named "Promoter", it will get "Promoter" as its type by using this option).

10.3.4 Removing annotations

Annotations can be hidden using the **Annotation Types** preferences in the **Side Panel** to the right of the view (see section 10.3.1). In order to completely remove the annotation:

right-click the annotation | Delete Annotation ( )

If you want to remove all annotations of one type:

right-click an annotation of the type you want to remove | Delete | Delete Annotations of Type "type"

If you want to remove all annotations from a sequence:

right-click an annotation | Delete | Delete All Annotations

The removal of annotations can be undone using Ctrl + Z or Undo ( ) in the Toolbar.

If you have more sequences (e.g. in a sequence list, alignment or contig), you have two additional options:

right-click an annotation | Delete | Delete All Annotations from All Sequences

right-click an annotation | Delete | Delete Annotations of Type "type" from All Sequences

10.4 Element information

The normal view of a sequence (by double-clicking) shows the annotations as boxes along the sequence, but often there is more information available about sequences. This information is

available through the **Element info** view.

To view the sequence information:

> **Select a sequence in the Navigation Area 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 10.19.

Figure 10.19: *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 (section **??**) 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.

10.5 View as text

A sequence can be viewed as text without any layout and text formatting. This displays all the information about the sequence in the GenBank file format. To view a sequence as text:

> **Select a sequence in the Navigation Area 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 10.4.)

In the **Side Panel**, you find a search field for searching the text in the view.

10.6 Sequence Lists

The **Sequence List** is a file containing a number of sequences. Having sequences in a sequence list can help organizing sequence data. A Sequence List can be displayed in a graphical sequence view or in a tabular format. The two different views of the same sequence list are shown in split screen in figure 10.20.

The **graphical view of sequence lists** is almost identical to the view of single sequences (see section 10.1). The main difference is that you now can see more than one sequence in the same view, and additionally have a few extra options for sorting, deleting and adding sequences:

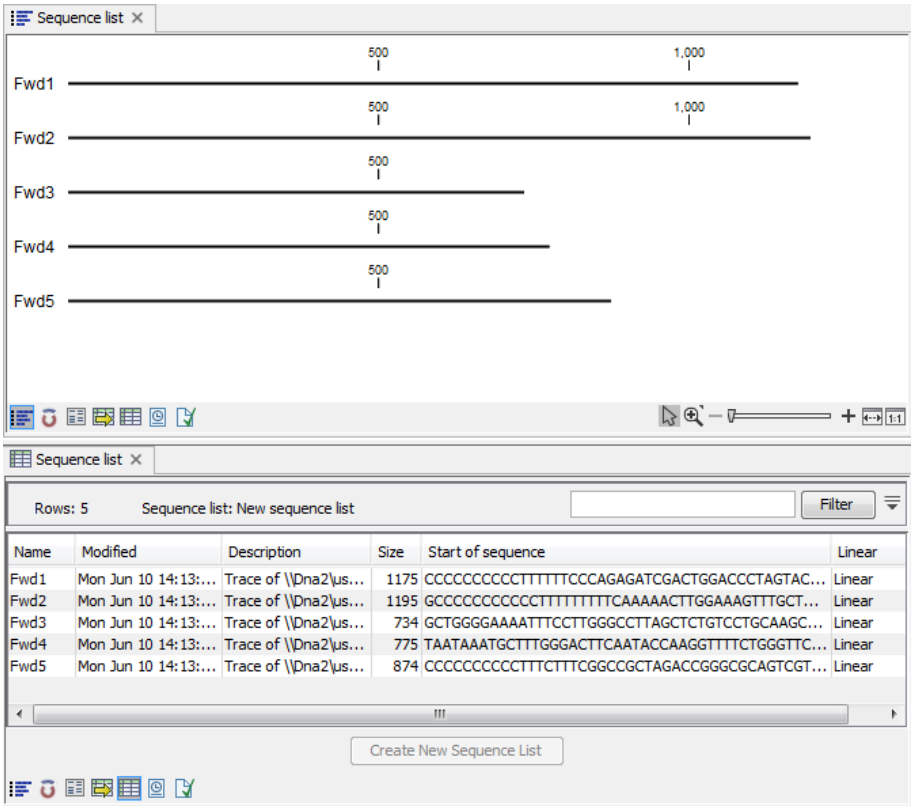


Figure 10.20: *A sequence list containing multiple sequences can be viewed in either a table or in a graphical sequence list. The graphical view is useful for viewing annotations and the sequence itself, while the table view provides other information like sequence lengths, and the number of sequences in the list (number of Rows reported).*

- To add extra sequences to the list, right-click an empty (white) space in the view, and select **Add Sequences**.
- To delete a sequence from the list, right-click the sequence's name and select **Delete Sequence**.
- To sort the sequences in the list, right-click the name of one of the sequences and select **Sort Sequence List by Name** or **Sort Sequence List by Length**.
- To rename a sequence, right-click the name of the sequence and select **Rename Sequence**.

Each sequence in the **table sequence list** is displayed with:

- Name
- Accession
- Description
- Modification date
- Length
- First 50 residues

The number of sequences in the list is reported as the number of Rows at the top of the table view. Adding and removing sequences from the list is easy: adding is done by dragging the sequence from another list or from the Navigation Area and drop it in the table. To delete sequences, simply select them and press **Delete** ( ). To extract a sequence from a sequence list, drag the sequence directly from the table into the Navigation Area. Another option is to extract all sequences found in the list using the **Extract Sequences** tool. A description of how to use the **Extract Sequences** tool can be found in section 31.1.

Sequence lists are generated automatically when you import files containing more than one sequence. They may also be created as the output from particular Workbench tool, including database searches.

You can create a subset of a Sequence List: select the relevant sequences, right-click on the selected elements and choose **Create New Sequence List** from the drop down menu. This will generate a new sequence list that only includes the selected sequences.

A **Sequence List** can also be created from single sequences or by merging already existing sequence lists with the Workbench. To do this, select two or more sequences or sequence lists in the Navigation Area, right click on the selected elements and choose

New | Sequence List ( )

Alternatively, you can launch this tool via the "File" menu system.

This opens the **Sequence List** Wizard (figure 10.21). The dialog allows you to select more sequences to include in the list, or to remove already chosen sequences from the list.

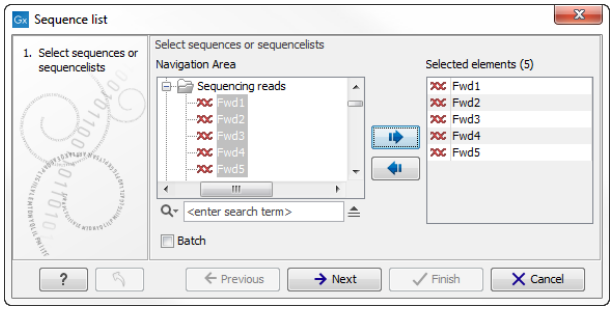


Figure 10.21: *A Sequence List dialog.*

If you are trying to create a new sequence list from a mixture of paired and unpaired datasets, a warning message will let you know that the resulting sequence list will be set as unpaired (figure 10.22).

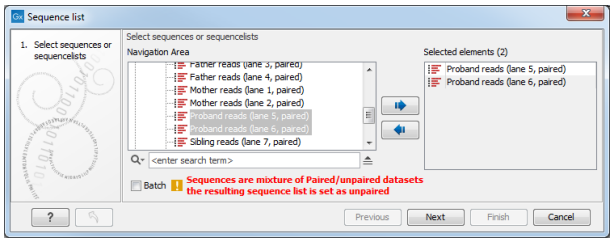


Figure 10.22: *A warning appears when trying to create a new sequence list from a mixture of paired and unpaired datasets.*

This warning also appears when trying to create a Sequence List out of paired reads lists for which the the Minimum and Maximum distances are different between lists. If that is the case,

distances can be edited to be similar for all lists that needs to be merged in a new one.

For this, open all Sequence Lists one after the other and click on the Show Element Info icon at the bottom of the view (figure 10.23). Edit the distances by clicking on the button "Edit" next to the entry "Paired status" and click OK. Save the Sequence lists with the edited Paired statuses before attempting to create a merged sequence List. This final list's status will be set as Paired reads.

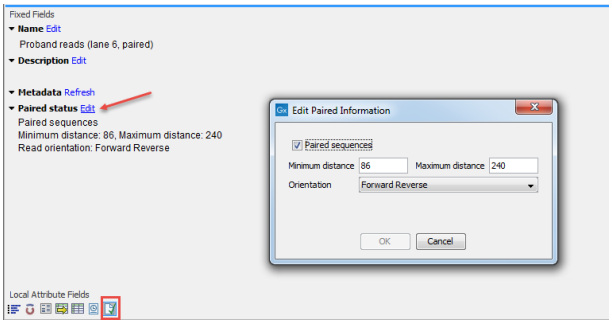


Figure 10.23: *Edit the Minimum and Maximum distances of several sequence lists to be able to merge them into one.*

Chapter 11

Viewing structures

Contents

Proteins are amino acid polymers that are involved in all aspects of cellular function. The structure of a protein is defined by its particular amino acid sequence, with the amino acid sequence being referred to as the primary protein structure. The amino acids fold up in local structural elements; helices and sheets, also called the secondary structure of the protein. These structural elements are then packed into globular folds, known as the tertiary structure or the three dimensional structure.

In order to understand protein function it is often valuable to see the three dimensional structure of the protein. This is possible when the structure of the protein has been resolved and published. Structure files are usually deposited in the Protein Data Bank (PDB) `http://www.rcsb.org/`, where the publicly available protein structure files can be searched and downloaded. The vast majority of the protein structures have been determined by X-ray crystallography (88%) while the rest of the structures predominantly have been obtained by Nuclear Magnetic Resonance techniques.

In addition to protein structures, the PDB entries also contain structural information about molecules that interact with the protein, such as nucleic acids, ligands, cofactors, and water.

There are also entries, which contain nucleic acids and no protein structure. The **3D Molecule Viewer** in the *Biomedical Genomics Workbench* is an integrated viewer of such structure files.

The **3D Molecule Viewer** offers a range of tools for inspection and visualization of molecular structures:

- Automatic sorting of molecules into categories: Proteins, Nucleic acids, Ligands, Cofactors, Water molecules
- Hide/unhide individual molecules from the view
- Four different atom-based molecule visualizations
- Backbone visualization for proteins and nucleic acids
- Molecular surface visualization
- Selection of different color schemes for each molecule visualization
- Customized visualization for user selected atoms
- Alignment of protein structures
- Browse amino acids and nucleic acids from sequence editors started from within the 3D Molecule Viewer
- Link a sequence or alignment to a protein structure
- Transfer annotations between the linked sequence and the structure

11.1 Importing molecule structure files

There are two ways to create a Molecule Project (3D molecule view) in *Biomedical Genomics Workbench*.

- Import a Protein Data Bank (PDB) file from your own file system using Standard Import (section 6.1)
- The Link Variants to 3D Protein Structure tool (section 23.9) creates a Molecule Project showing a 3D view of the variant's consequence on the protein structure.

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 11.1).

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 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.

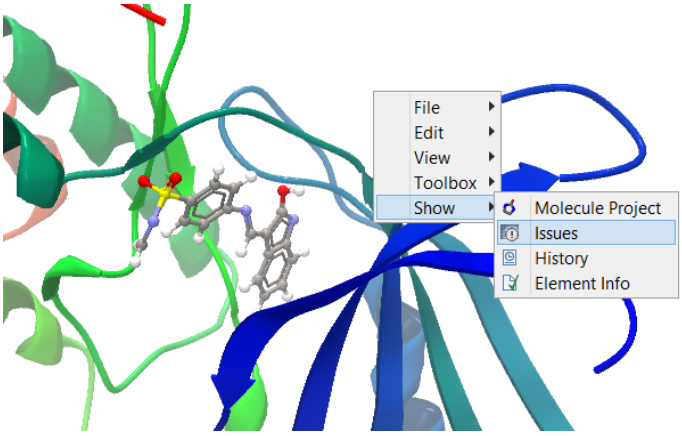


Figure 11.1: *At the bottom of the Molecule Project it is possible to switch to the "Show Issues" view by clicking on the "table-with-exclamation-mark" icon.*

11.2 Viewing molecular structures in 3D

An example of a 3D structure that has been opened as a **Molecule Project** is shown in figure 11.2.

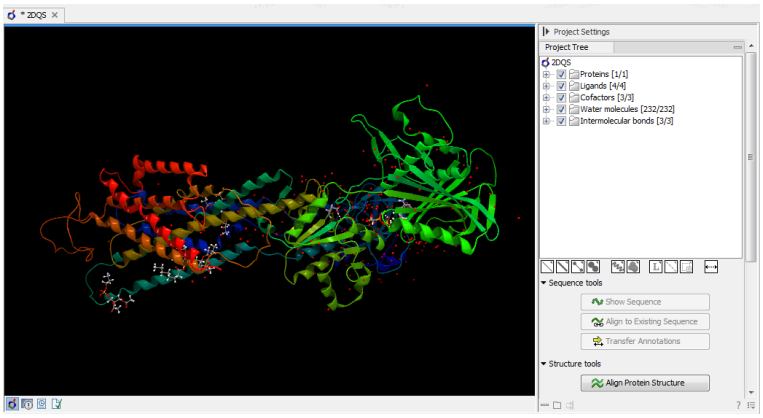


Figure 11.2: *3D view of a calcium ATPase. All molecules in the PDB file are shown in the Molecule Project. The Project Tree in the right side of the window lists the involved molecules.*

Moving and rotating The molecules can be rotated by holding down the left mouse button while moving the mouse. The right mouse button can be used to move the view.

Zooming can be done with the scroll-wheel or by holding down both left and right buttons while moving the mouse up and down.

All molecules in the **Molecule Project** are listed in categories in the **Project Tree**. The individual molecules or whole categories can be hidden from the view by un-cheking the boxes next to them.

It is possible to bring a particular molecule or a category of molecules into focus by selecting the molecule or category of interest in the **Project Tree** view and double-click on the molecule or category of interest. Another option is to use the zoom-to-fit button (⇤⇥) at the bottom of the **Project Tree** view.

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.

11.3 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.

11.3.1 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 11.3.1)

It is possible to adjust the opacity of a surface by adjusting the transparency slider at the bottom of the menu.

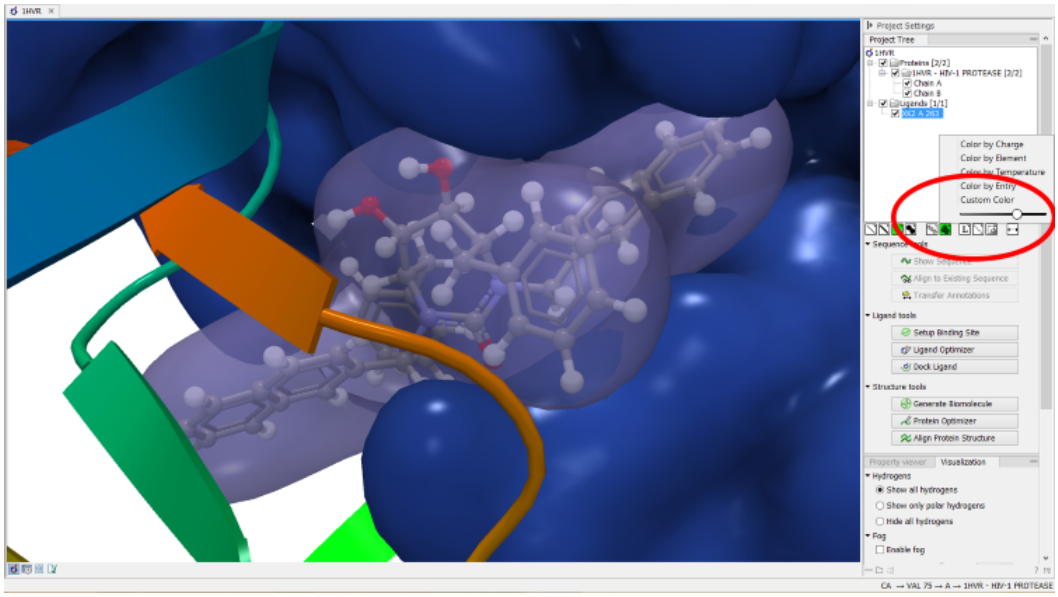


Figure 11.3: *Transparent surfaces*

Notice that visual artifacts may appear when rotating a transparent surface. These artifacts disappear as soon as the mouse is released.

Labels

(L)

Labels can be added to the molecules in the view by selecting an entry in the Project Tree and clicking the label button at the bottom of the Project Tree view. The color of the labels can be adjusted from the context menu by right clicking on the selected entry (which must be highlighted in blue first) or on the label button in the bottom of the Project Tree view (see figure 11.4).

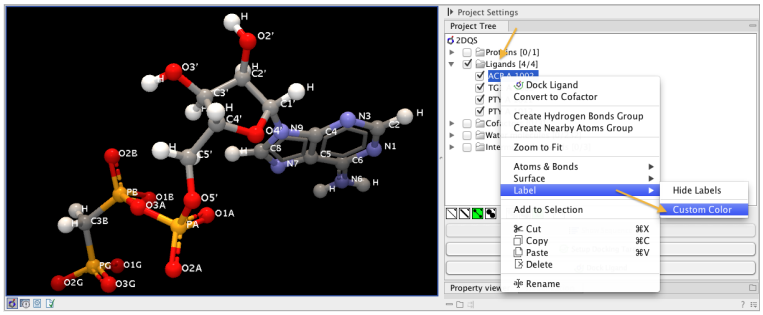


Figure 11.4: *The color of the labels can be adjusted in two different ways. Either directly using the label button by right clicking the button, or by right clicking on the molecule or category of interest in the Project Tree.*

- For proteins and nucleic acids, each residue is labeled with the PDB name and number.
- For ligands, each atom is labeled with the atom name as given in the input.
- For cofactors and water, one label is added with the name of the molecule.
- For atom groups including protein atoms, each protein residue is labeled with the PDB name and number.
- For atom groups not including protein atoms, each atom is labeled with the atom name as given in the input.

Labels can be removed again by clicking on the label button.

Hydrogen bonds

()

The Show Hydrogen Bond visualization style may be applied to molecules and atom group entries in the project tree. If this style is enabled for a project tree entry, hydrogen bonds will be shown to all other currently visible objects. The hydrogen bonds are updated dynamically: if a molecule is toggled off, the hydrogen bonds to it will not be shown.

It is possible to customize the color of the hydrogen bonds using the context menu.

Create atom group

()

Often it is convenient to use a unique visualization style or color to highlight a particular set of atoms, or to visualize only a subset of atoms from a molecule. This can be achieved by creating an atom group. Atom groups can be created based on atoms selected in the 3D view or entries selected in the Project Tree. When an atom group has been created, it appears as an entry in the Project Tree in the category "Atom groups". The atoms can then be hidden or shown, and the visualization changed, just as for the molecule entries in the Project Tree.

Note that an atom group entry can be renamed. Select the atom group in the Project Tree and invoke the right-click context menu. Here, the Rename option is found.

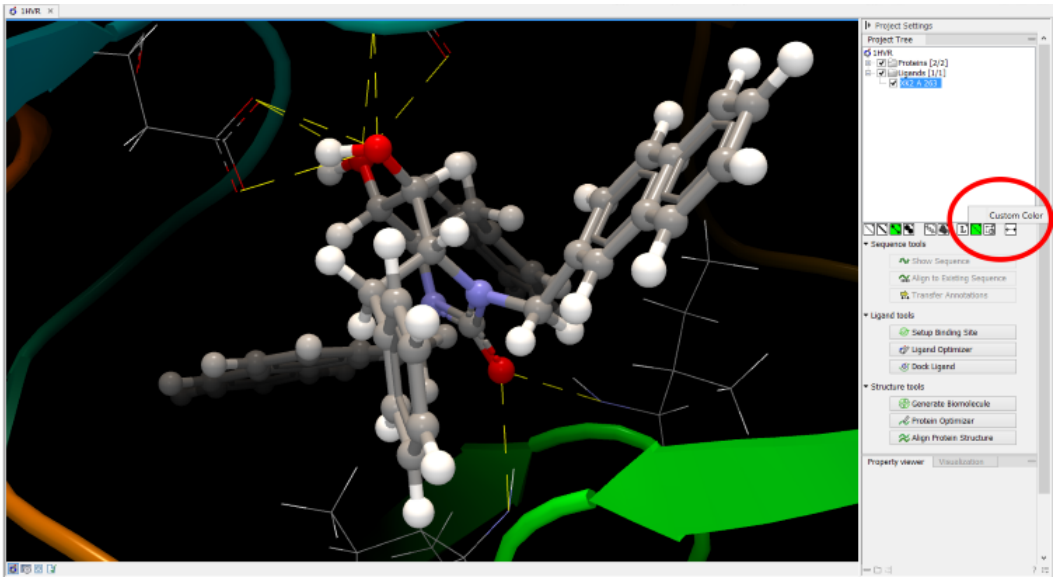


Figure 11.5: *The hydrogen bond visualization setting, with custom bond color*

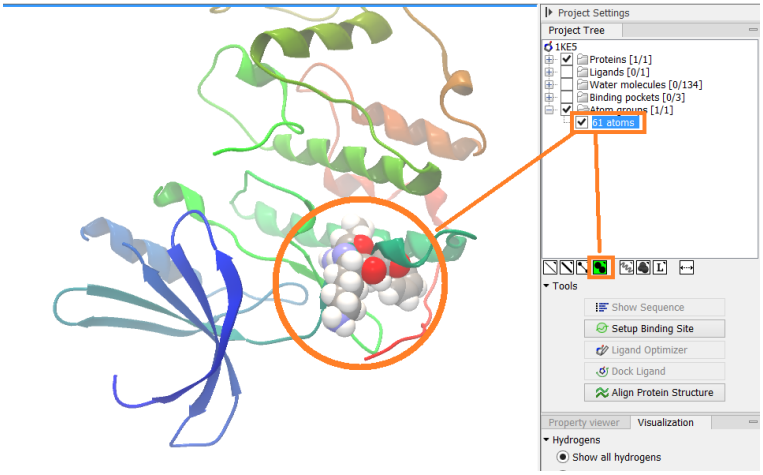


Figure 11.6: *An atom group that has been highlighted by adding a unique visualization style.*

Create atom group based on atoms selected in 3D view

When atoms are selected in the 3D view, brown spheres indicate which atoms are included in the selection. The selection will appear as the entry "Current" in the Selections category in the Project Tree.

Once a selection has been made, press the "Create Atom Group" button and a context menu will show different options for creating a new atom group based on the selection:

- Selected Atoms. Creates an atom group containing exactly the selected atoms (those indicated by brown spheres). If an entire molecule or residue is selected, this option is not displayed.
- Selected Residue(s)/Molecules. Creates an atom group that includes all atoms in the selected residues (for entries in the protein and nucleic acid categories) and molecules (for the other categories).
- Nearby Atoms. Creates an atom group that contains residues (for the protein and nucleic

acid categories) and molecules (for the other categories) within 5 Å of the selected atoms. Only atoms from currently visible Project Tree entries are considered.

- Hydrogen Bonded Atoms. Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) that have hydrogen bonds to the selected atoms. Only atoms from currently visible Project Tree entries are considered.

There are several ways to select atoms in the 3D view:

- Double click to select. Click on an atom to select it. When you double click on an atom that belongs to a residue in a protein or in a nucleic acid chain, the entire residue will be selected. For small molecules, the entire molecule will be selected.
- Adding atoms to a selection. Holding down Ctrl while picking atoms, will pile up the atoms in the selection. All atoms in a molecule or category from the Project Tree, can be added to the "Current" selection by choosing "Add to Current Selection" in the context menu. Similarly, entire molecules can be removed from the current selection via the context menu.
- Spherical selection. Hold down the shift-key, click on an atom and drag the mouse away from the atom. Then a sphere centered on the atom will appear, and all atoms inside the sphere, visualized with one of the all-atom representations will be selected. The status bar (lower right corner) will show the radius of the sphere.
- Show Sequence. Another option is to select protein or nucleic acid entries in the Project Tree, and click the "Show Sequence" button found below the Project Tree (section 11.4.1). A split-view will appear with a sequence editor for each of the sequence data types (Protein, DNA, RNA) (figure 11.7). If you then select residues in the sequence view, the backbone atoms of the selected residues will show up as the "Current" selection in the 3D view and the Project Tree view. Notice that the link between the 3D view and the sequence editor is lost if either window is closed, or if the sequence is modified.
- Align to Existing Sequence. If a single protein chain is selected in the Project Tree, the "Align to Existing Sequence" button can be clicked (section 11.4.2). This links the protein sequence with a sequence or sequence alignment found in the Navigation Area. A split-view appears with a sequence alignment where the sequence of the selected protein chain is linked to the 3D structure, and atoms can be selected in the 3D view, just as for the "Show Sequence" option.

Create atom group based on entries selected in the Project Tree

Select one or more entries in the Project Tree, and press the "Create Atom Group" button, then a context menu will show different options for creating a new atom group based on the selected entries:

- Nearby Atoms. Creates an atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) within 5 Å of the selected entries. Only atoms from currently visible Project Tree entries are considered.
- Hydrogen Bonded Atoms. Creates at atom group that contains residues (for the protein and nucleic acid categories) and molecules (for the other categories) that have hydrogen

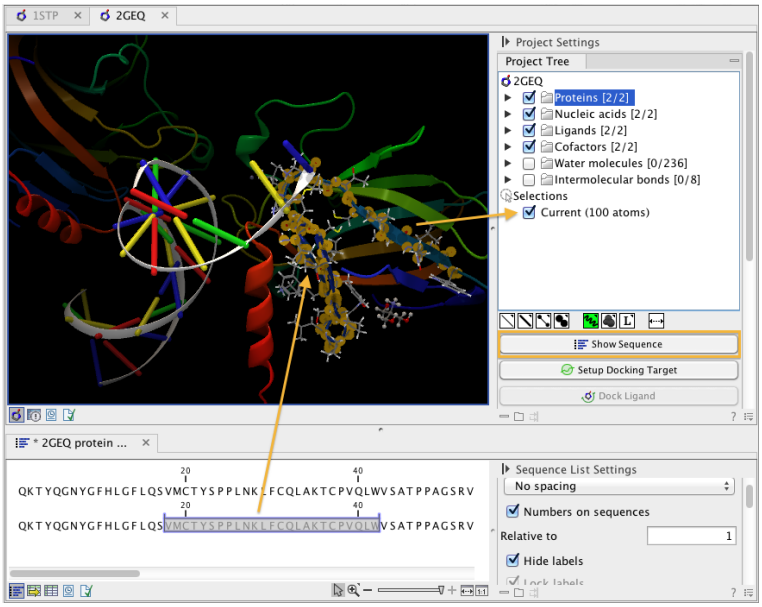


Figure 11.7: *The protein sequence in the split view is linked with the protein structure. This means that when a part of the protein sequence is selected, the same region in the protein structure will be selected.*

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.

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 11.8). Double-clicking an entry in the Project Tree will have the same effect.

11.3.2 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.

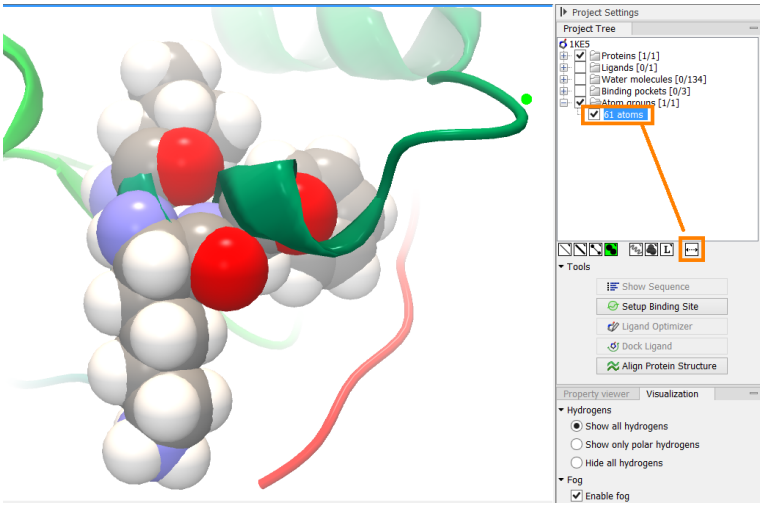


Figure 11.8: *The "Fit to screen" button can be used to bring a particular molecule or category of molecules in focus.*

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 11.4.1.
- **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 11.4.2.
- **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 11.4.3.
- **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 11.5.

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.

For atoms in protein models created by tools in the workbench, the following extra information is given:

- **Temperature** For structure models, the temperature value is an estimate of local structure uncertainty. The three aspects contributing to the assigned atom temperature is also listed, and described in section 23.8. The temperature value is a measure of uncertainty or disorder in the atom position; the higher the number, the higher the disorder.
- **Occupancy** For modeled structures and atoms, the occupancy is set to zero.

If an atom is selected, the Property view will be frozen with the details of the selected atom shown. If then a second atom is selected (by holding down Ctrl while clicking), the distance between the two selected atoms is shown. If a third atom is selected, the angle for the second atom selected is shown. If a fourth atom is selected, the dihedral angle measured as the angle between the planes formed by the three first and three last selected atoms is given.

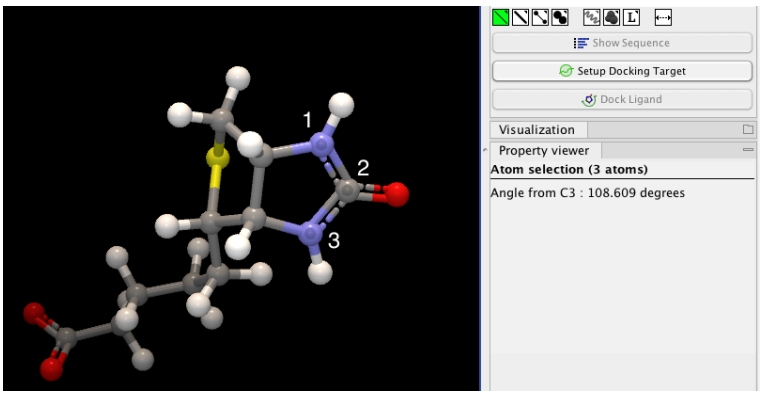


Figure 11.9: *Selecting two, three, or four atoms will display the distance, angle, or dihedral angle, respectively.*

If a molecule is selected in the Project Tree, the Property view shows information about this molecule. Two measures are always shown:

- **Atoms** Number of atoms in the molecule.
- **Weight** The weight of the molecule in Daltons.

Visualization settings

Under "Visualization" five options exist:

- **Hydrogens** Hydrogen atoms can be shown (Show all hydrogens), hidden (Hide all hydrogens) or partially shown (Show only polar hydrogens).
- **Fog** "Fog" is added to give a sense of depth in the view. The strength of the fog can be adjusted or it can be disabled.
- **Clipping plane** This option makes it possible to add an imaginary plane at a specified distance along the camera's line of sight. Only objects behind this plane will be drawn. It is possible to clip only surfaces, or to clip surfaces together with proteins and nucleic acids. Small molecules, like ligands and water molecules, are never clipped.
- **3D projection** The view is opened up towards the viewer, with a "Perspective" 3D projection. The field of view of the perspective can be adjusted, or the perspective can be disabled by selecting an orthographic 3D projection.
- **Coloring** The background color can be selected from a color palette by clicking on the colored box.

Snapshots of the molecule visualization To save the current view as a picture, right-click in the **View Area** and select "File" and "Export Graphics". Another way to save an image is by pressing the "Graphics" button in the Workbench toolbar ( ). Next, select the location where you wish to save the image, select file format (PNG, JPEG, or TIFF), and provide a name, if you wish to use another name than the default name.

You can also save the current view directly on data with a custom name, so that it can later be applied (see section 4.6).

11.4 Tools for linking sequence and structure

The *Biomedical Genomics 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 11.4.3).

11.4.1 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 11.10). 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 11.3.1. 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 11.4.2).

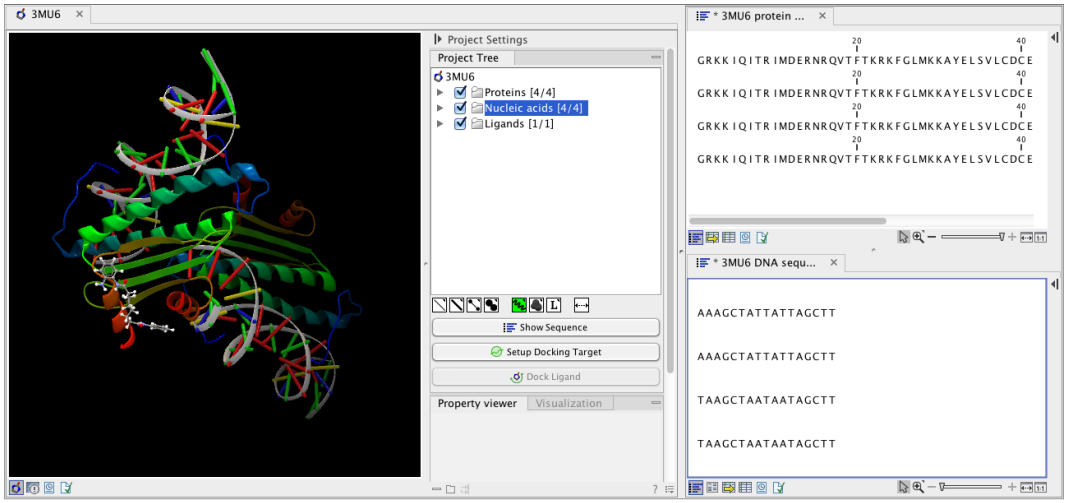


Figure 11.10: *Protein chain sequences and DNA sequences are shown in separate views.*

11.4.2 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 11.4.1) or Align to Existing Sequence.

Select a single protein chain in the project tree (see figure 11.11). Pressing "Align to Existing Sequence" then opens a Navigation Area browser, where it is possible to select one or more Sequence, Sequence Lists, or Alignments, to link with the selected protein chain.

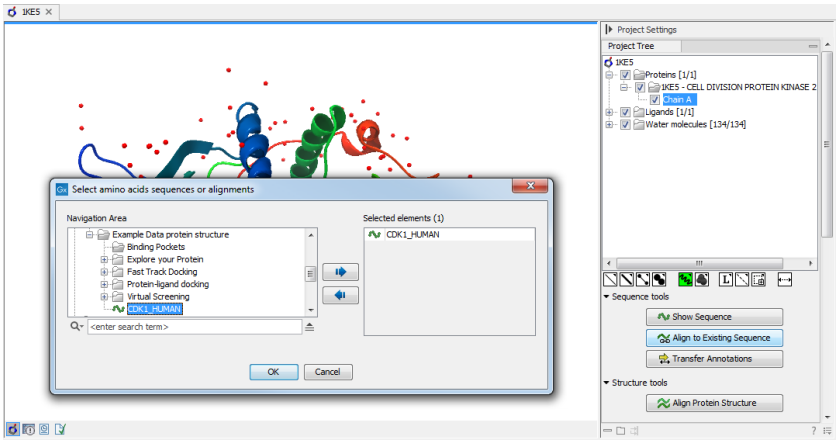


Figure 11.11: *Select a single protein chain in the Project Tree and invoke "Align to Existing Sequence".*

If the sequences or alignments already contain a sequence identical to the protein chain selected in the Molecule Project (i.e. same name and amino acid sequence), this sequence is linked to the protein structure. If no identical sequence is present, a sequence is extracted from the protein structure (as for Show Sequence - section 11.4.1), and a sequence alignment is created between this sequence and the sequences or alignments selected from the Navigation Area.

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 11.4.3). Notice, that the link will be broken if

either the sequence or the 3D protein chain is modified.

11.4.3 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 10.3 and more about atom groups in section 11.3.1.

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 11.4.1) or by mapping to an existing sequence or sequence alignment using the 'Align to Existing Sequence' button (see section 11.4.2).

Invoke the Transfer Annotations dialog by selecting a linked protein chain in the Project Tree and press 'Transfer Annotations' (see figure 11.12).

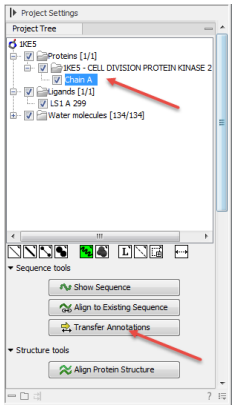


Figure 11.12: *Select a single protein chain in the Project Tree and invoke "Transfer Annotations".*

The dialog contains two tables (see figure 11.13). The left table shows all atom groups in the Molecule Project, with at least one atom on the selected protein chain. The right table shows all annotations present on the linked sequence. While the Transfer Annotations dialog is open, it is not possible to make changes to neither the sequence nor the Molecule Project, however, changes to the visualization styles are allowed.

How to undo annotation transfers

In order to undo operations made using the Transfer Annotations dialog, the dialog must first be closed. To undo atom groups added to the structure, activate the 3D view by clicking in it and press Undo in the Toolbar. To undo annotations added to the sequence, activate the sequence view by clicking in it and press Undo in the Toolbar.

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

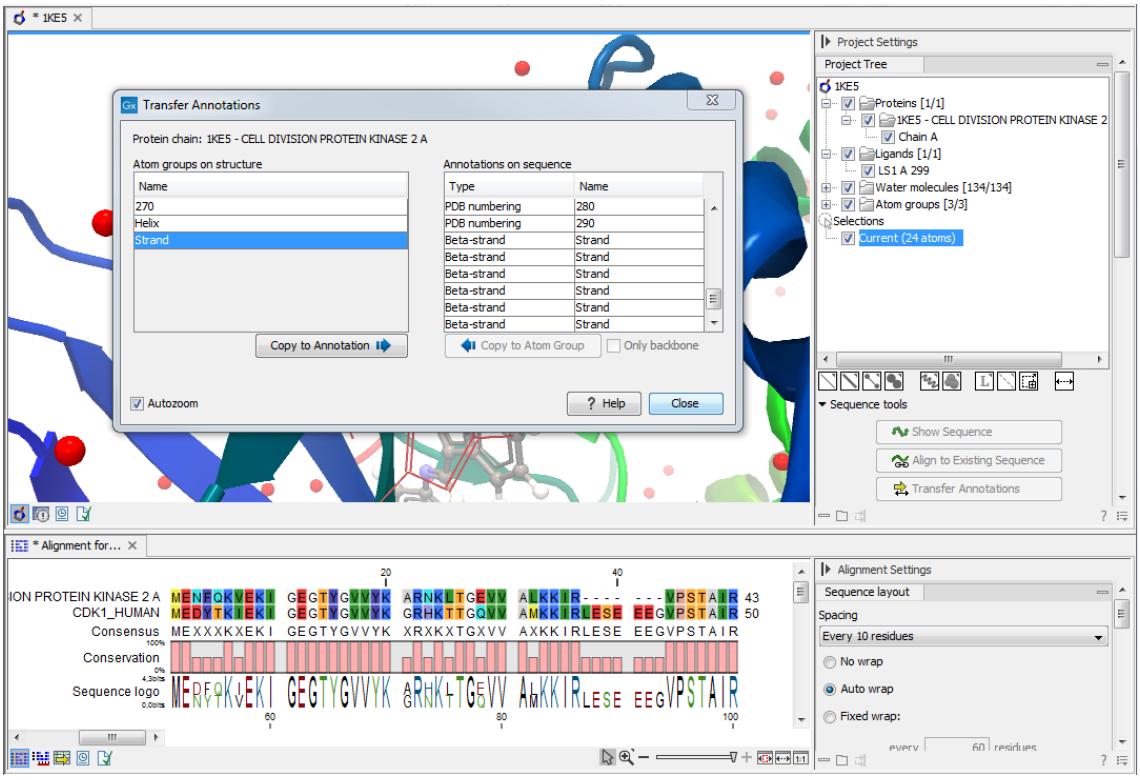


Figure 11.13: *The Transfer Annotations dialog allow you to select annotations listed in the two tables, and copy them from structure to sequence or vice versa.*

copy (see figure 11.14).

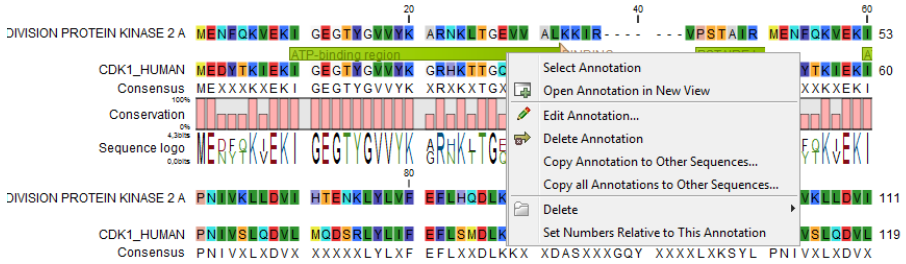


Figure 11.14: *Copy annotations from sequences in the alignment to the sequence linked to the 3D view.*

11.5 Protein structure alignment

The Align Protein Structure tool allows you to compare a protein or binding pocket in a **Molecule Project** with proteins from other **Molecule Projects**. The tool is invoked using the () Align Protein Structure action from the **Molecule Project Side Panel**. This action will open an interactive dialog box (figure 11.15). 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.

11.5.1 The Align Protein Structure dialog box

The dialog box contains three fields:

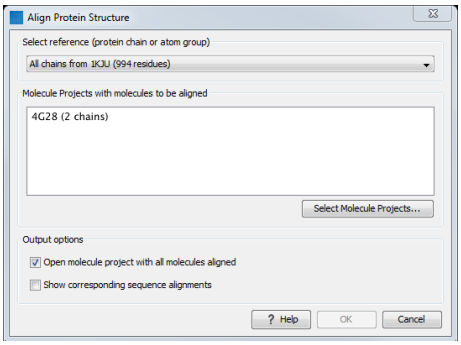


Figure 11.15: *The Align Protein Structure dialog box.*

- **Select reference (protein chain or atom group)** This drop-down menu shows all the protein chains and residue-containing atom groups in the current **Molecule Project**. If an atom group is selected, the structural alignment will be optimized in that area. The 'All chains from *Molecule Project* option will create a global alignment to all protein chains in the project, fitting e.g. a dimer to a dimer.
- **Molecule Projects with molecules to be aligned** One or more **Molecule Projects** containing protein chains may be selected.
- **Output options** The default output is a single **Molecule Project** containing all the input projects rotated onto the coordinate system of the reference. Several alignment statistics, including the RMSD, TM-score, and sequence identity, are added to the **History** of the output **Molecule Project**. Additionally, a sequence alignments of the aligned structures may be output, with the sequences linked to the 3D structure view.

11.5.2 Example: alignment of calmodulin

Calmodulin is a calcium binding protein. It is composed of two similar domains, each of which binds two calcium atoms. The protein is especially flexible, which can make structure alignment challenging. Here we will compare the calcium binding loops of two calmodulin crystal structures – PDB codes 1A29 and 4G28.

Initial global alignment The 1A29 project is opened and the Align Protein Structure dialog is filled out as in figure 11.15. 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 11.16. 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).

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**

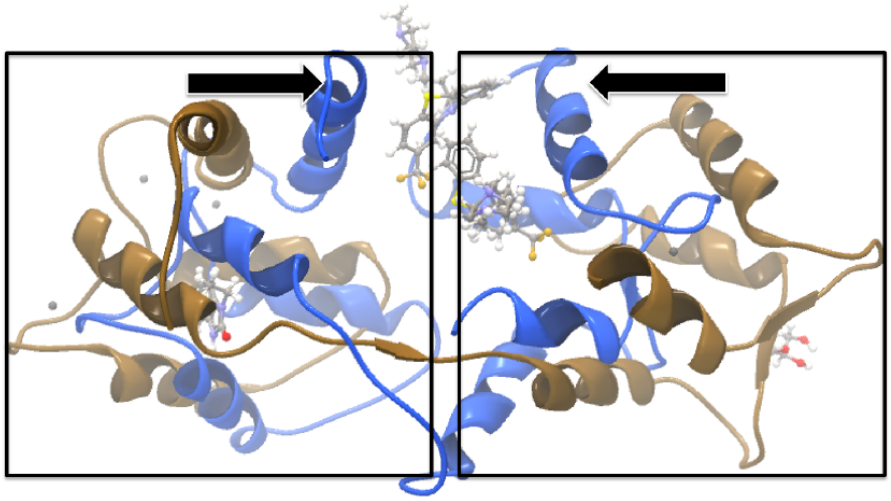


Figure 11.16: *Global alignment of two calmodulin structures (blue and brown). The two domains of calmodulin (shown within black boxes) can undergo large changes in relative orientation. In this case, the different orientation of the domains in the blue and brown structures makes a good global alignment 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.*

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 11.17). Using the new atom group as the reference in the alignment dialog leads to the alignment shown in figure 11.18. 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.

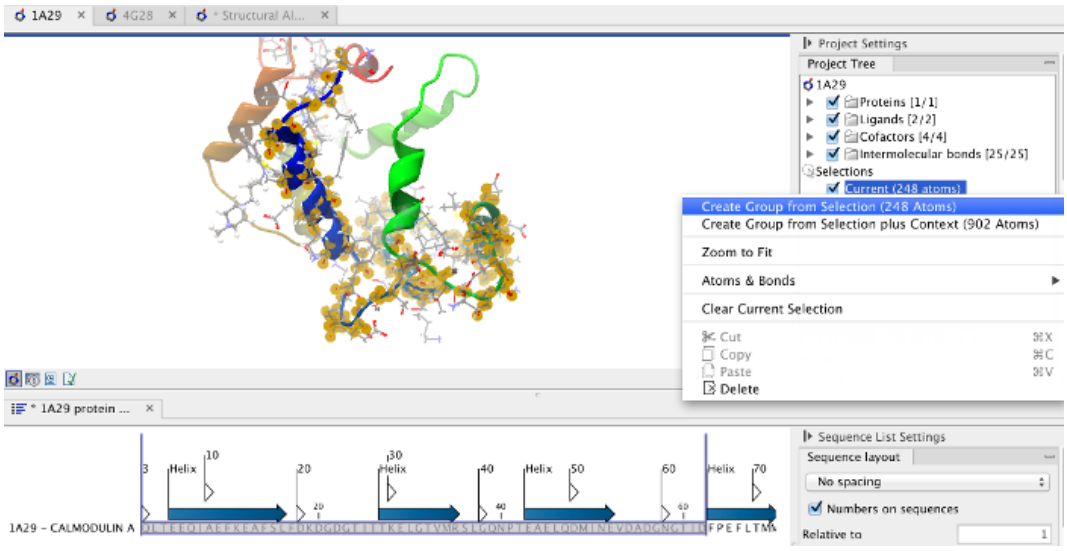


Figure 11.17: *Creation of an atom group containing the N-terminal domain of calmodulin.*

Aligning a binding site Two bound calcium atoms, one from each calmodulin structure, are shown in the black box of figure 11.18. 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,

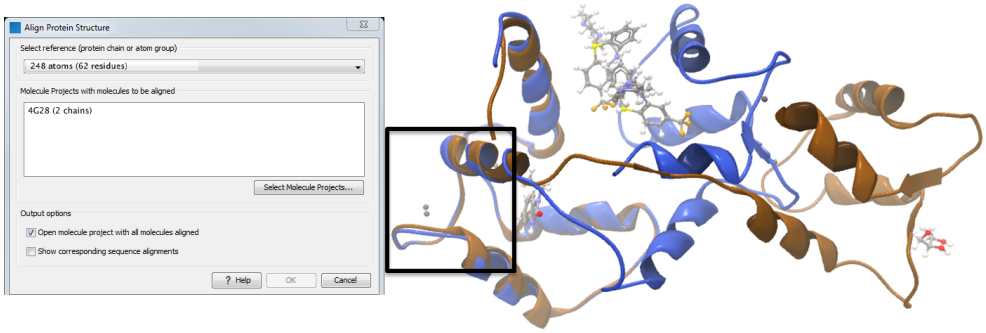


Figure 11.18: *Alignment of the same two calmodulin proteins as in figure 11.16, but this time with a focus on the N-terminal domain. The blue and brown structures are now well-superimposed in the N-terminal region. The black box encloses two calcium atoms that are bound to the structures.*

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 11.19.

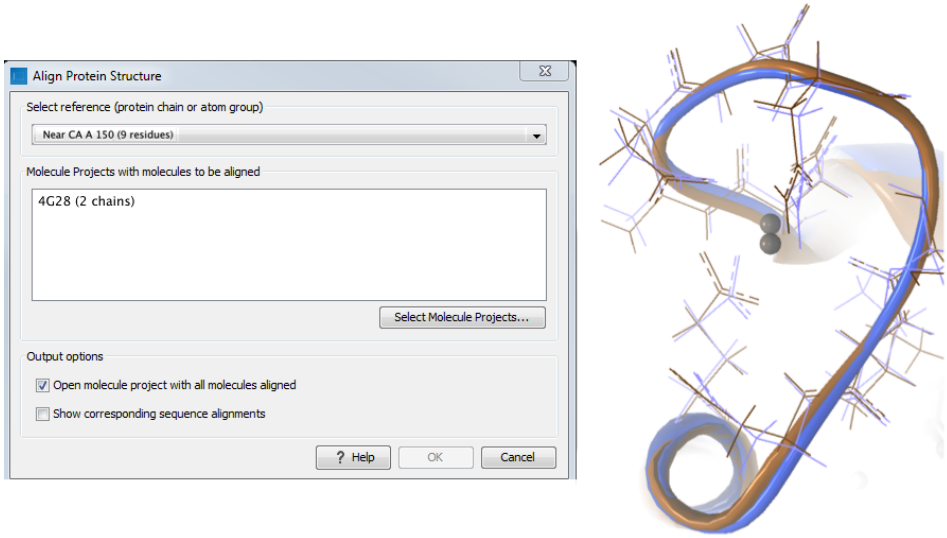


Figure 11.19: *Alignment of the same two calmodulin domains as in figure 11.16, but this time with a focus on the calcium atom within the black box of figure 11.18. The calcium atoms are less than 1 Å apart – compatible with thermal motion encoded in the atoms' temperature factors.*

11.5.3 The Align Protein Structure algorithm

Any approach to structure alignment must make a trade-off between alignment length and alignment accuracy. For example, is it better to align 200 amino acids at an RMSD of 3.0 Å or 150 amino acids at an RMSD of 2.5 Å? The Align Protein Structure algorithm determines the answer to this question by taking the alignment with the higher TM-score. For an alignment focused on a protein of length L, this is:

$$\text{TM-score} = \frac{1}{L}\sum_i \frac{1}{1+\frac{d_i}{d(L)}^2}$$

where i runs over the aligned pairs of residues, d_i is the distance between the i^{th} such pair, and $d(L)$ is a normalization term that approximates the average distance between two randomly chosen points in a globular protein of length L [Zhang and Skolnick, 2004]. A perfect alignment has a TM-score of 1.0, and two proteins with a TM-score $>$0.5 are often said to show structural homology [Xu and Zhang, 2010].

The Align Protein Structure Algorithm attempts to find the *structure alignment* with the highest 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).

Part III

Applications - ready-to-use workflows

Chapter 12

Ready-to-Use Workflows descriptions and guidelines

Contents

Biomedical Genomics Workbench contains several ready-to-use workflows that support analysis of cancer data, but also analysis of hereditary diseases and other conditions that are best studied using family analysis.

Before running an application workflow, it is important to prepare the sequencing reads as explained in the following diagram (figure 12.1).

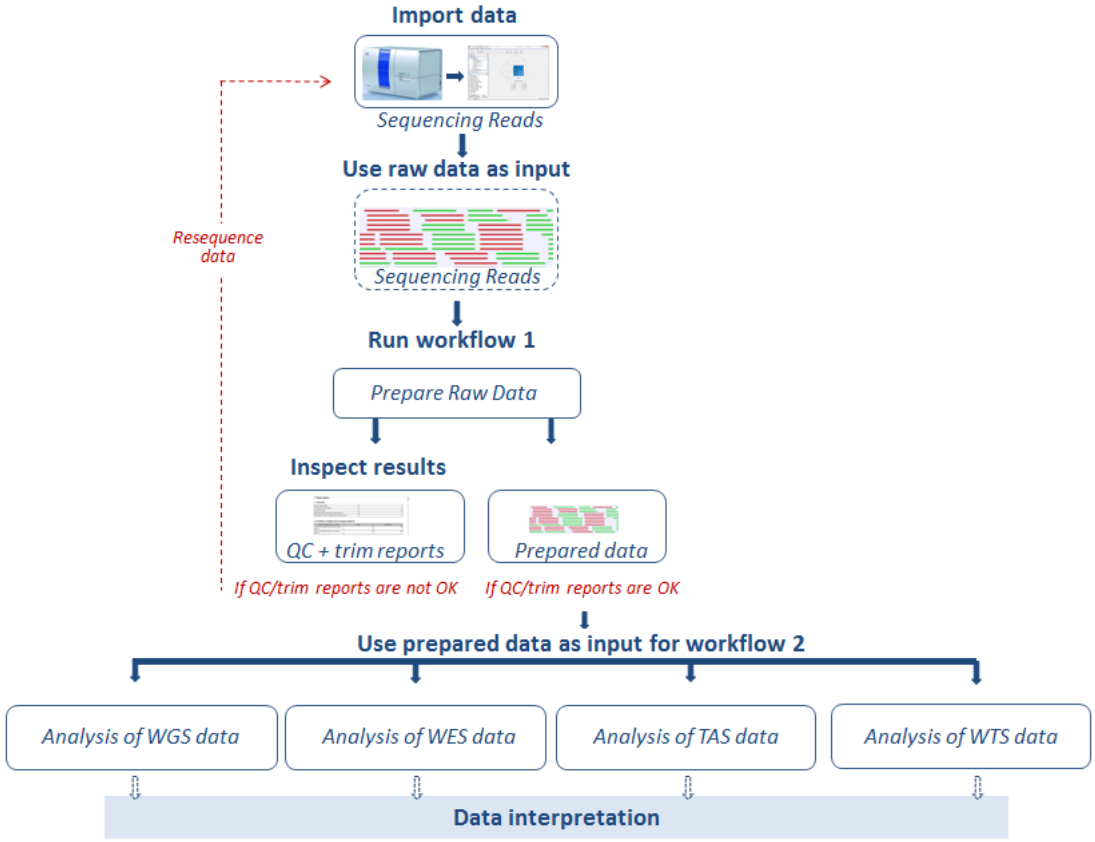


Figure 12.1: *From sequencing reads to data interpretation.*

The workflows are specific to the type of data used as input: Whole Genome Sequencing (WGS), Whole Exome Sequencing (WES), Targeted Amplicon Sequencing (TAS) and Whole Transcriptome Sequencing (WTS). For each of the first three categories, WGS, WES, and TAS, workflows exist that can be used for general identification and annotation of variants irrespective of disease, these workflows are found in a folder called **General Analysis**. In folders called **Somatic Cancer**, you can find workflows that are specific for cancer research. Finally, you will find a folder under each of the WGS, WES, and TAS applications, that is labeled **Hereditary Disease**. The workflows found in this folder can be used for studying variants that cause rare diseases or hereditary diseases (HD).

The ready-to-use workflows found under each of the first three applications have similar names (with the only difference that "WGS", "WES", or "TAS", or have been added after the name). However, some of the workflows have been tailored to the individual applications with parameter settings that have been adjusted to fit e.g. the expected differences in coverage between the different application types. We therefore recommend that you use the ready-to-use workflow that is found under the relevant application heading.

12.1 General Workflow

The General workflows are universal workflows in the sense that they can be used independently of the disease that is being studied. Two workflows exist in this category:

- *Annotate Variants*: Annotates variants with gene names, conservation scores, amino acid changes, and information from relevant databases.
- *Identify Known Variants in One Sample*: Maps sequencing reads and looks for the presence or absence of user-specified variants in the mapping.

12.2 Somatic Cancer

The Somatic Cancer ready-to-use workflows are workflows that have been tailored to cancer research. In this category it is possible to find e.g. workflows that can compare variants in matched tumor normal pairs. The workflows found in the Somatic Cancer category, use the "Low Frequency Variant Detection" for variant calling. The advantages of using this variant caller when analyzing cancer data are that 1) it does not take ploidy into consideration, and 2) it is particularly good at picking up low frequency variants in contrast to the other variant callers.

The workflows that are available in this category are:

- *Filter Somatic Variants*: Removes variants outside the target region (only targeted experiments) and common variants present in publicly available databases. Annotates with gene names, conservation scores, and information from relevant databases.
- *Identify Somatic Variants from Tumor Normal Pair*: Removes germline variants by referring to the control sample read mapping, removes variants outside the target region (in case of a targeted experiment), and annotates with gene names, conservation scores, amino acid changes, and information from relevant databases.
- *Identify Variants*: Calls variants in the mapped and locally realigned reads, removes false positives and, in case of a targeted experiment, removes variants outside the targeted region. Variant calling is performed with the Low Frequency Variant Detection tool.

12.3 Hereditary Disease

The third category found under each of the three applications WGS, WES, and TAS are the Hereditary Disease workflows that have been developed to support identification of disease causing mutations in families.

Hereditary diseases can be non-cancer related diseases, such as inherited heart diseases or familial hypercholesterolemia, or it can be inherited cancers such as hereditary colorectal cancer or hereditary breast cancer. In addition to the hereditary diseases, family analysis can help researchers identify rare disease causing mutations that can be:

- a new mutation, also known as a de novo mutation, that is only present in a child and not in any of the parents
- a combination of events that occur in the same gene but at different positions in each of the parents, which is not disease causing by itself in either of the parents, but when both variants are found in a child, it becomes disease causing; this type of variant is known as a compound heterozygous variant.

A range of different workflows exist in this category that have been optimized for different purposes. In the current version of the *Biomedical Genomics Workbench* we offer workflows tailored to two family sizes, 1) a classical "Trio", consisting of a mother, father, and an affected child (the proband), and 2) a "Family of Four", which is mother, father, affected child, and either a sibling (in the workflows that detects rare diseases) or another affected family member (in the workflows that detect inherited diseases), that can be any affected relative such as a sibling, grand parent or the like. The workflows use the "Fixed Ploidy Variant Detection" tool, which is a variant caller that has been designed to call variants in samples with known ploidy from read mapping data. Workflows designed to detect rare variants can both pick up de novo variants as well as compound heterozygous variants. In addition to the Trio and Family of Four workflows, additional workflows exist that have been designed to pick up variants that are inherited from either the mother or the father.

The available workflows in this category are:

- *Filter Causal Variants*: Removes variants outside the target region (only targeted experiments) and common variants present in publicly available databases. Annotates with gene names, conservation scores, and information from relevant databases.
- *Identify Causal Inherited Variants in a Family of Four*: Identifies putative disease causing inherited variants by creating a list of variants present in all three affected individuals and subtracting all variants in the unaffected individual. The workflow includes a back-check for all family members.
- *Identify Causal Inherited Variants in a Trio*: Identifies putative disease causing inherited variants by creating a list of variants present in both affected individuals and subtracting all variants in the unaffected individual. The workflow includes a back-check for all family members
- *Identify Rare Disease Causing Mutations in a Family of Four*: Identifies *de novo* and compound heterozygous variants from an extended family of four, where the fourth individual is not affected.
- *Identify Rare Disease Causing Mutations in a Trio*: Identifies *de novo* and compound heterozygous variants from a Trio. The workflow includes a back-check for all family members.
- *Identify Variants (HD)*: Calls variants in the mapped and locally realigned reads, removes false positives and, in case of a targeted experiment, removes variants outside the targeted region. Variant calling is performed with the Fixed Ploidy Variant Detection tool.

Although each workflow design to analyze Hereditary Diseases is specific to the data used or the type of analysis, they share several tools and steps:

Below you can find a general description for how to run a workflow in the category "Hereditary diseases". In some workflows, such as the "Filter Causal Variants" workflows you will be asked about a variant track as input. Other workflows start with specifying a reads track. This is the case for all workflows that starts with "Identify Variants.." in the name.

Note that in case of workflows annotating variants using databases available for more than one population, you can select the population that matches best the population your samples are

derived from. This will be done in the wizard for populations from the 1000 Genomes Project, while Hapmap populations are specified with the **Data Management** ( ) function before starting the workflows (see section 13.1).

Select the variant track (figure 12.2). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the variant track name or click once on the file and then click on the arrow pointing to the right side in the middle of the wizard.

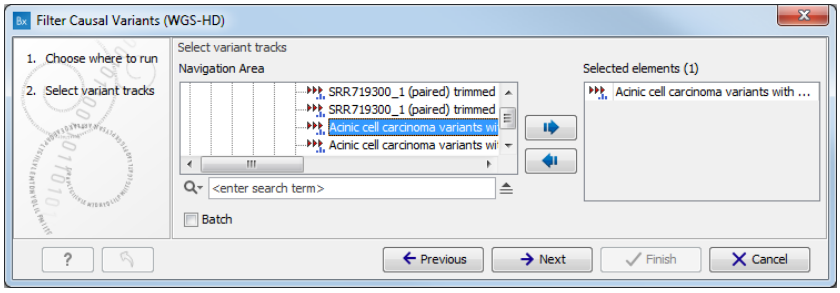


Figure 12.2: *Select the variant track from which you would like to filter somatic variants.*

Specify the sequencing reads for each family member (figure 12.3).

The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

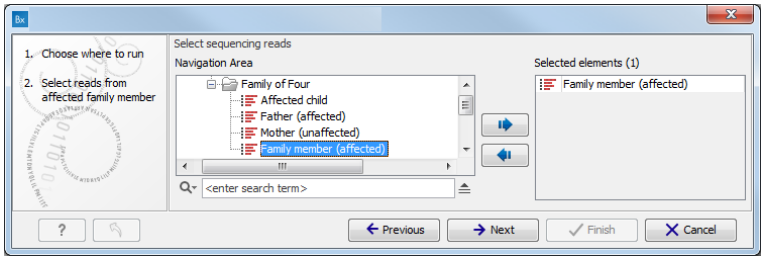


Figure 12.3: *Specify the sequencing reads for the appropriate family member.*

Specify the targeted region file (figure 12.4).

The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents. Remember that you have a hg38-specific BED file when using hg38 as reference, and hg19-specific BED file when using hg19 as reference.

Specify the affected child's gender for the Trio analysis (figure 12.5).

Some workflows contains a Trio Analysis and thus take the gender of the proband into account.

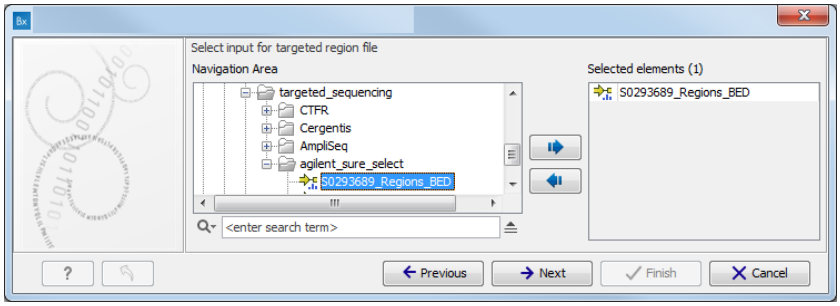


Figure 12.4: *Select the targeted region file you used for sequencing.*

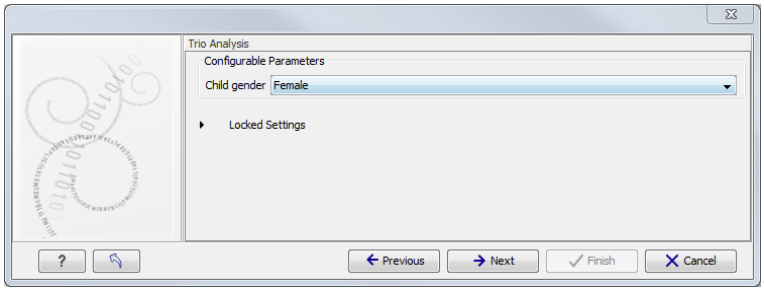


Figure 12.5: *Specify the proband's gender.*

Specify the parameters for the Fixed Ploidy Variant Detection tool (figure 12.6).

The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

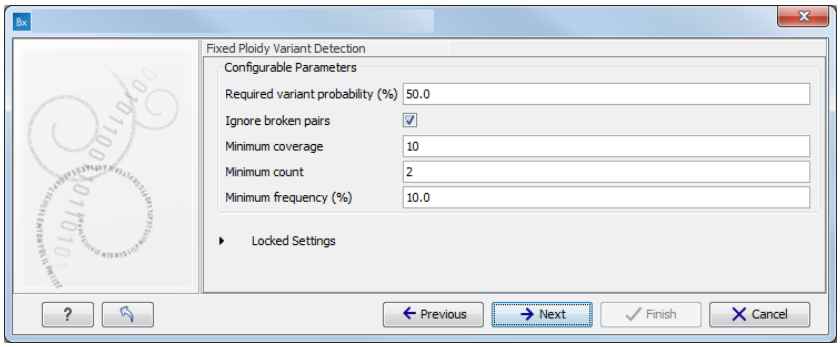


Figure 12.6: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called

might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

Specify the parameters for the QC for Target Sequencing tool (figure 12.7).

When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

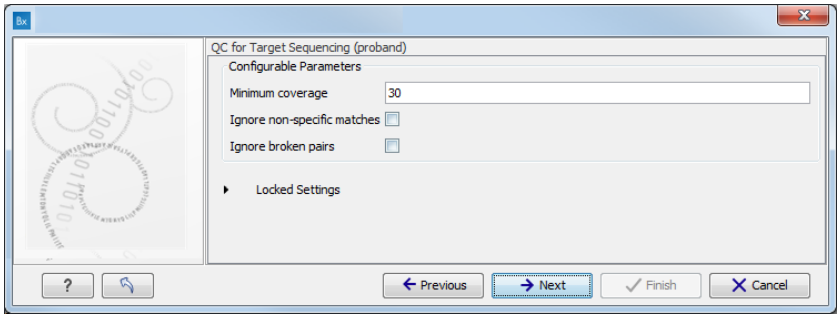


Figure 12.7: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

When asked for it, specify the targeted regions track (figure 12.8).

For more information about the tool, see section 20.1.

Map Reads to a reference (figure 12.9).

For this tool, the **Autodetect paired distances** settings is switched off in all Targeted Amplicon Sequencing workflows.

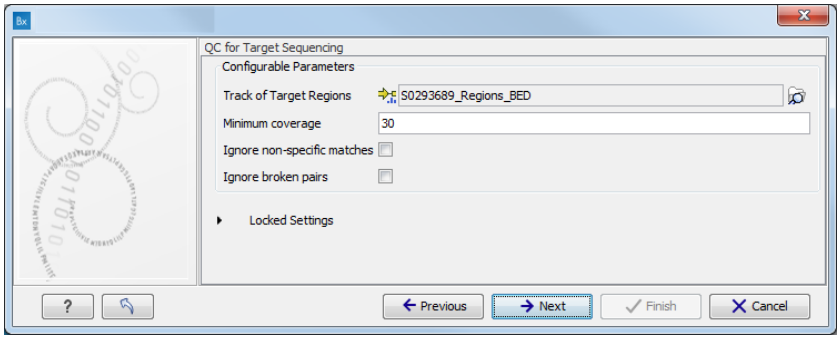


Figure 12.8: *Specify the parameters for the QC for Target Sequencing tool.*

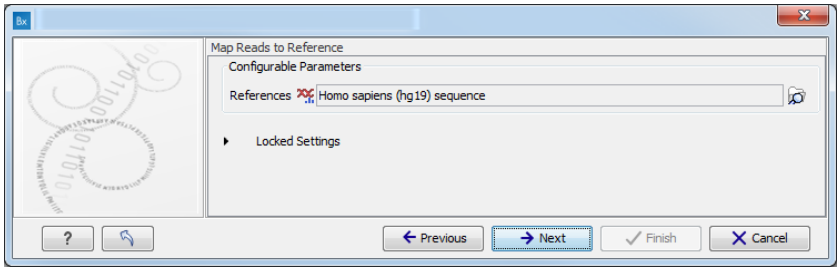


Figure 12.9: *Specify the parameters for the QC for Target Sequencing tool.*

Specify the target region for the Indels and Structural Variants tool (figure 12.10).

The targeted region file is a file that specifies which regions have been sequenced when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents. Remember that you have a hg38-specific BED file when using hg38 as reference, and hg19-specific BED file when using hg19 as reference.

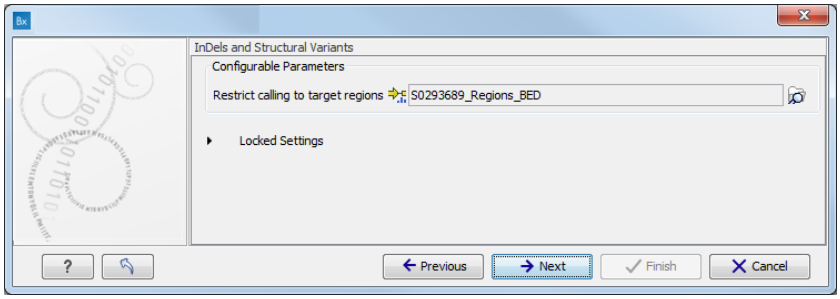


Figure 12.10: *Specify the parameters for the Indels and Structural Variants tool.*

Specify the relevant 1000 Genomes populations (figure 12.11)

Note: this window will appear in workflows that annotate variants with information from the 1000 Genomes project, unless you have already selected the relevant populations of interest in your reference data management prior to running the workflow.

Some wizard window will be called **Add Information from 1000 Genomes Project** or **Remove Variants found in the 1000 Genomes Project**. Specify the 1000 Genomes population that should be used to add or filter out variants found in the 1000 Genomes project. This can

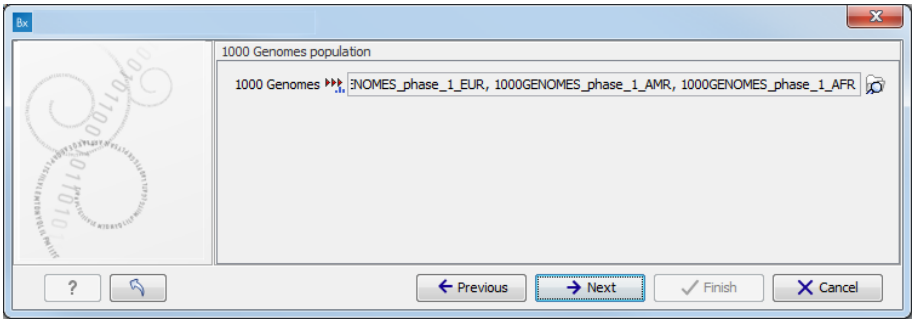


Figure 12.11: *Select the relevant 1000 Genomes population(s).*

be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

Specify the relevant Hapmap populations (figure 12.12)

Note: this window will appear in workflows that annotate variants with information from the Hapmap project, unless you have already selected the relevant populations of interest in your reference data management prior to running the workflow.

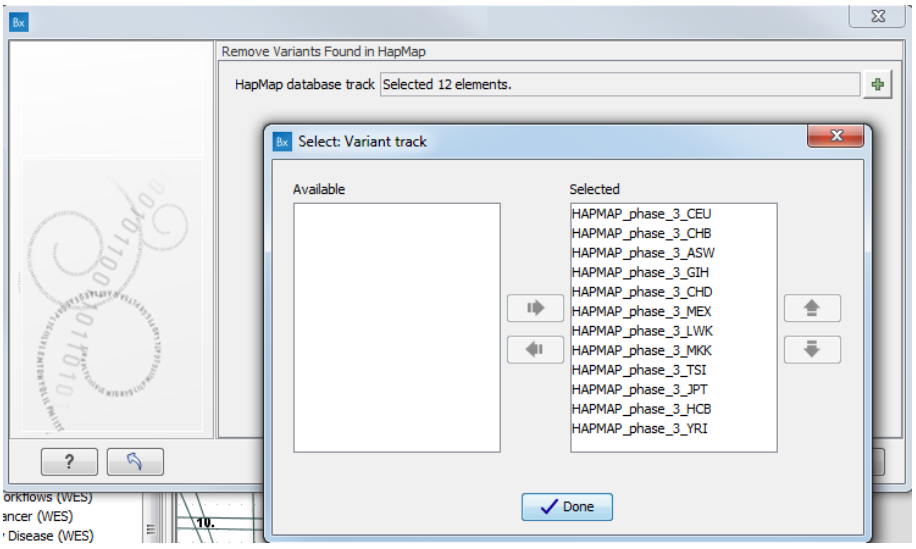


Figure 12.12: *Select the relevant Hapmap population(s).*

Some wizard window will be called **Add Information from the Hapmap project** or **Remove Variants found in Hapmap**. Specify the Hapmap population that should be used to add or filter out variants found in the Hapmap project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

Chapter 13

Reference data for ready-to-use workflows

Contents

The ready-to-use workflows rely on the presence of particular reference datasets. This reference data must be downloaded and configured before these workflows can be used. The Data Management tool (figure 13.1) in the workbench make it easy to download the necessary data such that the workflows can find and use it.

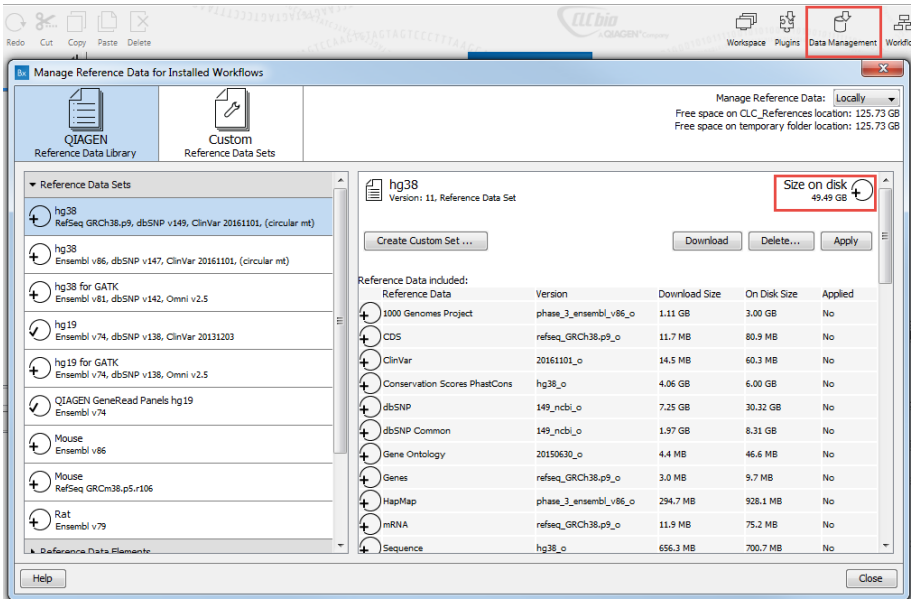


Figure 13.1: *Click on the Data Management button to find, download and customize reference data.*

This section covers the download and configurations needed to make available the reference data relevant to the *Biomedical Genomics Workbench*, including the human, mouse and rat genomes,

annotations and variants made available by a variety of databases. The total size of the reference data set you can download varies among the data set, and is indicated in the top right corner of the data set window (see the red highlight in figure 13.1). the size of the individuals files of the data set in indicated in the table below. The amount of time it will take to download this data depends on your network connection, but it can take several hours on slower connections.

Where reference data is downloaded from Reference data is provided by QIAGEN and the workbench is configured to download from QIAGEN by default. The location to download the data from can be seen in **Edit** | **Preferences** | **Advanced** as shown in figure 13.2.

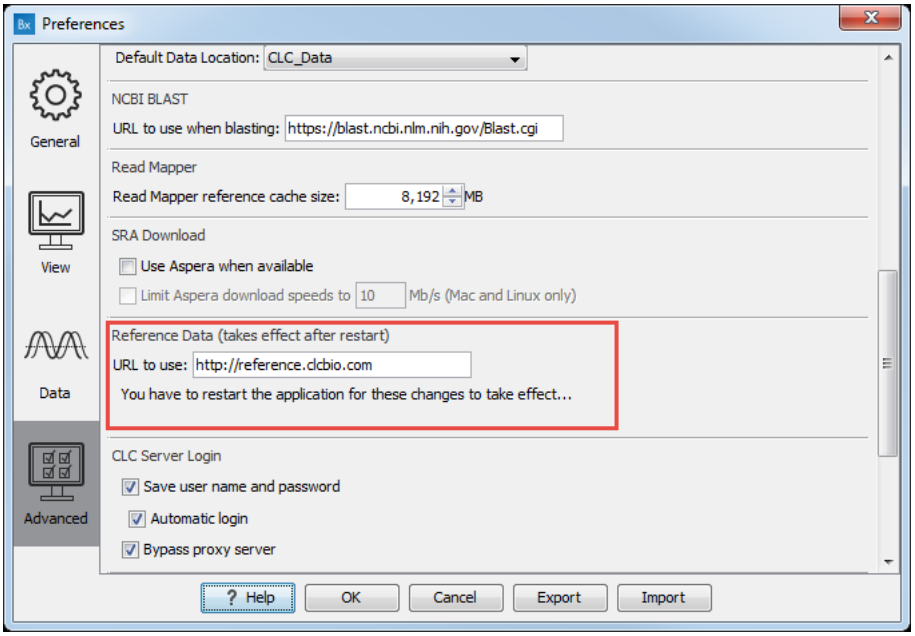


Figure 13.2: *The location where reference data is downloaded from can be seen in the Workbench Preferences. Generally this should not be altered except in the special case that the data from QIAGEN is being mirrored locally.*

Unless you are in the special circumstance that your system administrator has decided to mirror this data locally and wishes you to use that mirror of the data, you should **not** change this setting.

Where reference data is downloaded to The reference data that is downloaded will be stored in a folder called **CLC_References**. When the *Biomedical Genomics Workbench* is installed, such a folder is created on your file system under your home area. This folder is specified within the workbench as a **reference location**.

You can specify a different location to download reference data to. This is recommended if you do not have enough space in the area the workbench designates as the reference data location by default. To change the reference data location from within the **Navigation Area**:

> **Right-click on the folder "CLC_References"** | **Choose "Location"** | **Choose "Specify Reference Location"**

The new folder will also be called CLC_References, but will be located where you specify.

In more detail, this action results in the following:

- A folder called CLC_References is created in the location you specified, if a folder of this name did not already exist.
- The workbench sets this new location as the place to download reference data to and the place the ready-to-use workflows should look for reference data.

This action does **not**:

- Remove the old CLC_References folder.
- Remove the contents of the old CLC_References folder, such as previously downloaded data.

If you have previously downloaded data into the CLC_References folder with the old location, you will need to use standard system tools to delete this folder and/or its contents. If you would like to keep the reference data from the old location, you can move it, using standard system tools, into the new CLC_References folder that you just specified. This would save you needing to download it again.

Note! If you run out of space, and realize that the CLC_References should be stored somewhere else, you can do this by choosing a new location, then manually moving the already downloaded files to that new location, and restarting the workbench. The "downloaded references" file will then be updated with all the new references.

13.1 Download and configure reference data

The first time you open *Biomedical Genomics Workbench* you will be presented with the dialog box shown in figure 13.3, which informs you that data are available for download either to the local or server CLC_References repository. If you check the "Never show this dialog again" then subsequently you will only be presented with the dialog box when updated versions of the reference data are available.

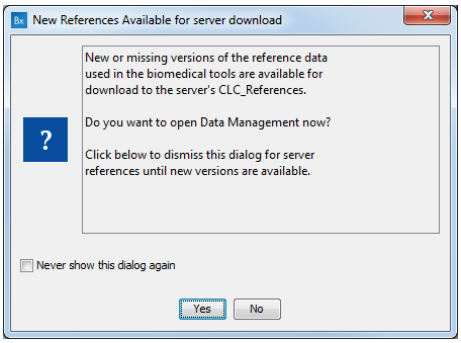


Figure 13.3: *Notification that new versions of the reference data are available.*

Click on the button labeled **Yes**. This will take you to the data manager shown in figure 13.4.

This wizard can also be accessed from the upper right corner of the *Biomedical Genomics Workbench* by clicking on **Data Management** ( ) (figure 13.5).

The "Manage Reference Data" wizard gives access to all the reference data that are used in the ready-to-use workflows and in the tutorials. From the wizard you can download and configure the reference data.

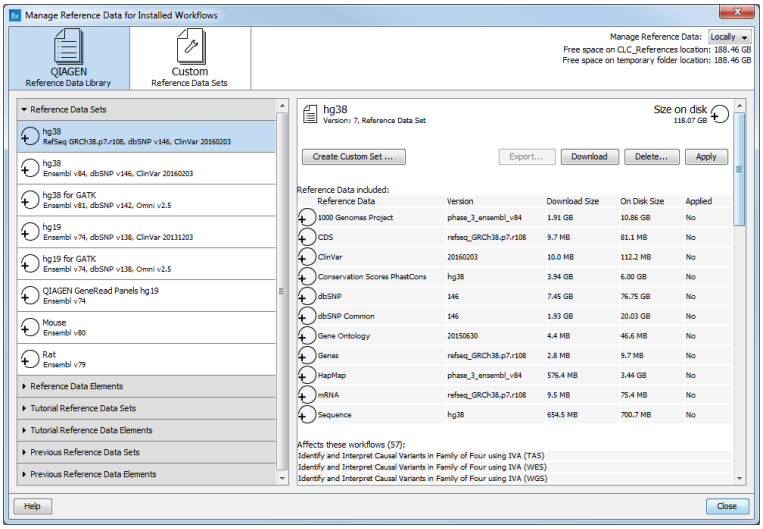


Figure 13.4: *The Manage Reference Data wizard gives access to the reference data that are required to be able to run the ready-to-use workflows.*

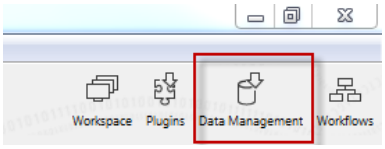


Figure 13.5: *Click on the button labeled "Data Management" to open the "Manage Reference Data" dialog where you can download and configure the reference data that are necessary to be able to run the ready-to-use-workflows.*

In the upper part of the wizard you can find two tiles called "QIAGEN Reference Data Library" ( ) and "Custom Reference Data Sets" ( ).

On the left hand side, you can use the drop-down menu to choose where you want to manage the reference data. If you choose "Locally", the Download, Delete and Apply buttons will work on the local reference data. If you choose "On Server" (only available if you are connected to the server), the buttons will work on the reference data on the server you are connected to(figure 13.6).

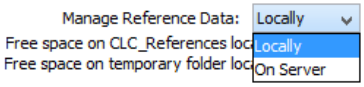


Figure 13.6: *Reference data can be available locally or on the server.*

You can also check how much free space is available for the Reference folder on your local disk or on the server. The drop-down menu also allows you to check which datasets have been downloaded locally or on the server. You can see this in the left panel of the reference data manager.

When on the "QIAGEN Reference Data Library" tile, we can see the list of all available references data under 6 headers: **Reference Data Sets** and **Reference Data Elements**, **Tutorial Reference Data Sets** and **Tutorial Reference Data Elements**, and **Previous Reference Data Sets** and **Previous Reference Data Elements**. Two icons indicate whether you have already downloaded your data in your Reference folder ( ) or not ( ).

When selecting a reference set or an element, the window on the right show the size of the folder as well as some complementary information about the reference database. For Reference

Data Sets, a table recapitulates the elements included in the set with their version number and respective size, as well as a list of the workflows affected by the set.

The **Reference Data Sets** available include hg19, hg38 (both an Ensembl and a RefSeq version), RefSeq, Mouse and Rat, a data set designed for QIAGEN Gene Reads Panels hg19, and two data sets for use with the GATK plugin.

We also offer access to **Tutorial Reference Data Sets** that are chromosome-specific and ready to us with some of our tutorials (http://www.qiagenbioinformatics.com/support/tutorials/).

The **Previous Reference Data Sets** folder contains older versions of the Reference Data Sets that have been replaced with newer one in the Reference Data Sets folder.

Each Reference Data Set is made of a compilation of Reference Data Elements. Downloading sets will automatically download the elements the set is made of, but you can also download elements individually under the **Reference Data Elements** folder.

Data that has not been downloaded yet is represented by a plus icon ( ). Select the set or element you would like to download, and click on the **Download** button. Once the data is downloading, the **Download** button fades out and you can check the progress of the downloading in the **Processes** tab below the toolbox (figure 13.7).

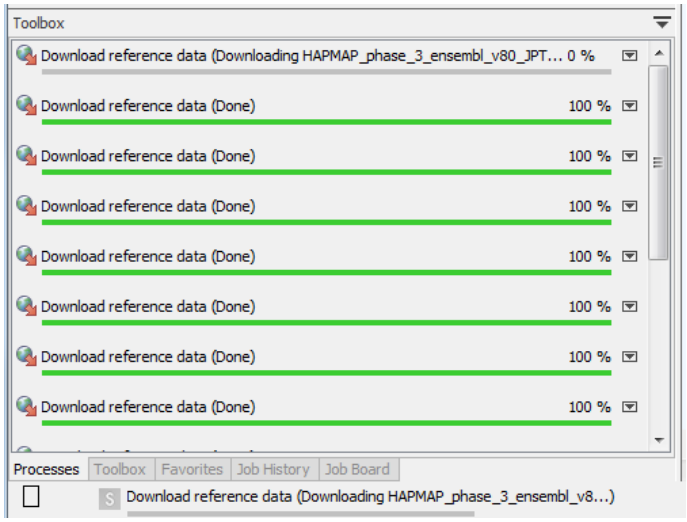


Figure 13.7: *Click on the info button to see the legal notice and license information.*

Once the reference data has been downloaded, the set or element is marked with a check icon ( ).

If you have finished downloading the appropriate Reference Data Set, click on the button labeled **Apply** and the workflows will automatically be configured with all the relevant reference data available. The information in the "Applied" column in the right panel of the reference data manager describes whether the dataset has been applied to the location specified in the drop-down menu. For example, a "Yes" in the "Applied" column when the drop-down menu is set to "On Server" means that the given data will be used from the server, when the affected workflows are run. This will be the case even if you choose execute the workflow locally (i.e. in the workbench). If the "Applied" column contains "Yes" when the drop-down menu is set to "Locally", this means the given data will be used from the local reference folder, when the affected workflows are run. This means that you will not be able to execute these workflows on the server (figure 13.8).

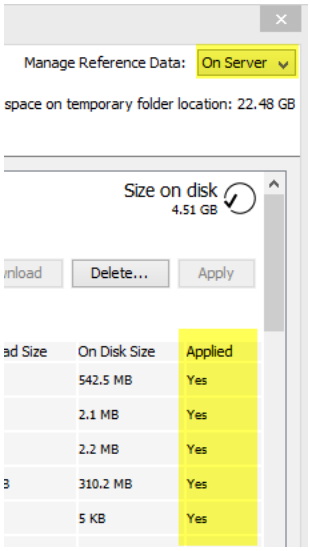


Figure 13.8: *Check where your reference data is applied by looking at the column "Applied" in the data set description.*

The Reference Data Sets also contain a **Create Custom Set ...** button that allows you to create your own set of reference data starting from an existing data set (see section 13.2).

The **Delete** button allows user to delete locally installed reference data, whereas only administrators are capable of deleting reference data installed on the server. This can be used if you suspect that a downloaded reference is corrupt, and needs to be re-downloaded, or if you need to clean up space, e.g. locally.

At the bottom of the wizard you can find:

- A button "Help" that links to the section in the *Biomedical Genomics Workbench* reference manual that describes the "Manage Reference Data" button.
- A button labeled "Close". Click on this to close the wizard.

13.2 Create a custom Reference Data Set

The Reference Data Sets also contain a **Create Custom Set ...** button that allows you to create your own set of reference data starting from an existing data set.

Clicking on this button will open a window (figure 13.9) where you can change the name of the new data set, the organism it represents, the chromosomal extension, and the annotation types used.

For each type of reference, a drop-down menu allows you to choose from the different versions available, as well as a custom option that allows you to import database and sequences saved in your Navigation Area (figure 13.10). This is useful when you have your own version of the reference data that you have imported in the workbench and that you would like to use rather than the currently available Reference Data Sets. The customs data sets are saved under the Custom Reference Data Sets tile. Do not forget to click on the button **Apply** if you wish to use this set for your workflows.

For references like the "1000 Genomes Project" and "HapMap" databases which contain more

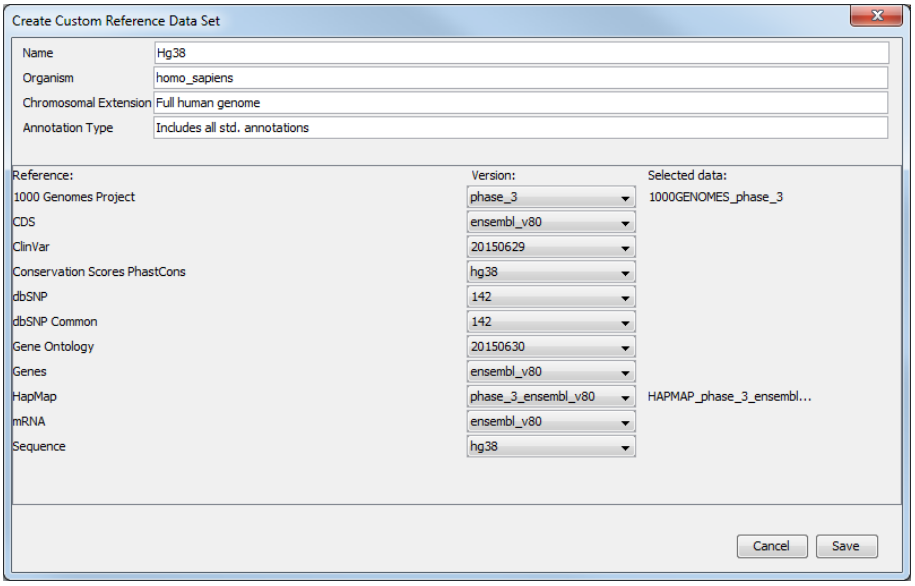


Figure 13.9: *Select the reference data elements you want to add to you custom reference data set.*

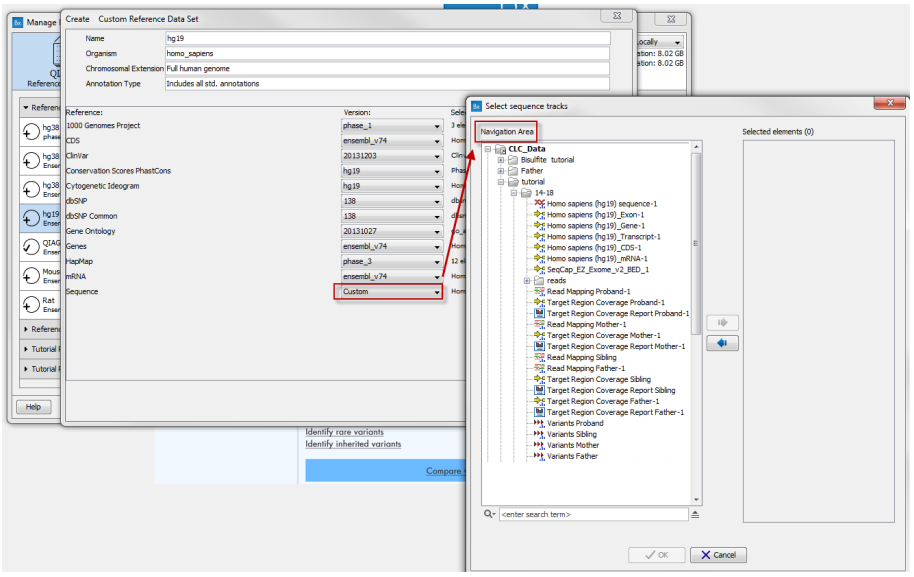


Figure 13.10: *Select data from your Navigation Area to create your own custom Reference Data Set.*

than one reference data file, the workflow will initially be configured with all the populations being available and you will be able to specify which reference data to use in the workflow wizard directly. But you can also modify a pre-existing Reference Data Set to contain only the population you want to work with. In the Data Management wizard, select the Reference Data Set you are interested in, click on **Create Custom Set**. Select the version of the 1000 genomes or Hapmap database you wish to work with (figure 13.11).

A pop-up window will open where you can select the population you want to work with. Alternatively, click on the option "custom" in lieu of version and choose from the CLC_References folder the population of your choice (figure 13.12).

Three letter codes are used to specify the population that the different reference data origin from (for example ASW = American's of African Ancestry in SW USA). For the phase 3 HapMap pop-

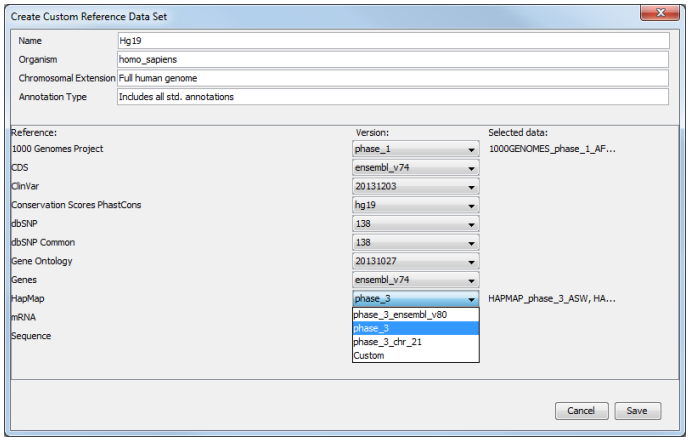


Figure 13.11: *Select the version of the 1000 genomes or Hapmap database you want to work with, or select the option "custom".*

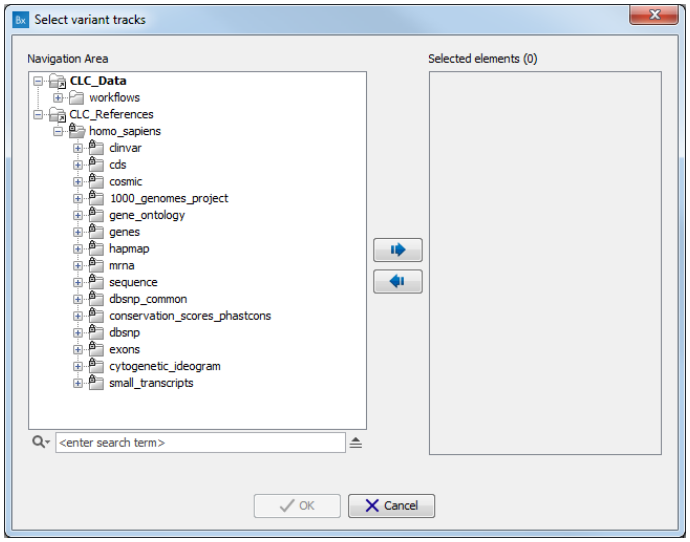


Figure 13.12: *Choosing the option "custom" allows you to choose your reference from the Navigation Area..*

ulation codes, please see http://www.sanger.ac.uk/resources/downloads/human/hapmap3.html and for the 1000 Genomes Project see http://www.1000genomes.org/category/frequently-asked-questions/population.

Note: Custom reference data sets specific to the workbench on which they are created, and will not appear in other workbenches connected to the same server.

13.3 Exporting reference data for use in external applications

The Reference Data Manager can export a reference data set. This can be very useful, e.g. when using the external applications framework. In this situation, one might be interested in using the reference data from the workbench in the external application, instead of having to obtain the relevant data by other means, and making them available to the external application.

To export a reference data set, the following conditions must be satisfied:

- The External Applications Client Plugin must be installed

- The workbench must be connected to a server
- There must be at least one import/export location on the server
- The reference data set to be exported must be fully downloaded to the **CLC_References** reference data location on the server
- The user is currently managing reference data on the server (as opposed to locally)

If those conditions are satisfied, there will be an enabled "Export..." button in the details pane for the reference data set (figure 13.13).

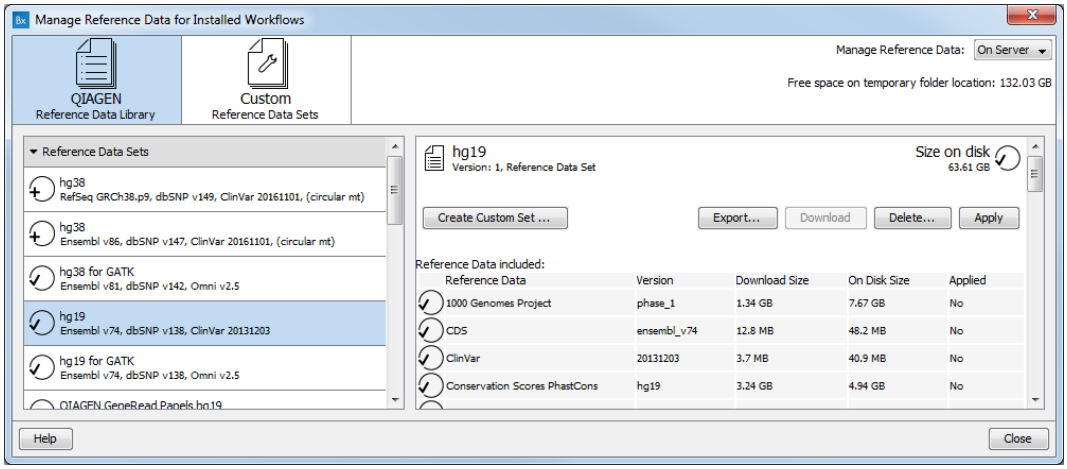


Figure 13.13: *The "Export..." button in the reference data manager.*

Pressing the export button results in the dialog for selecting the export location shown in figure 13.14.

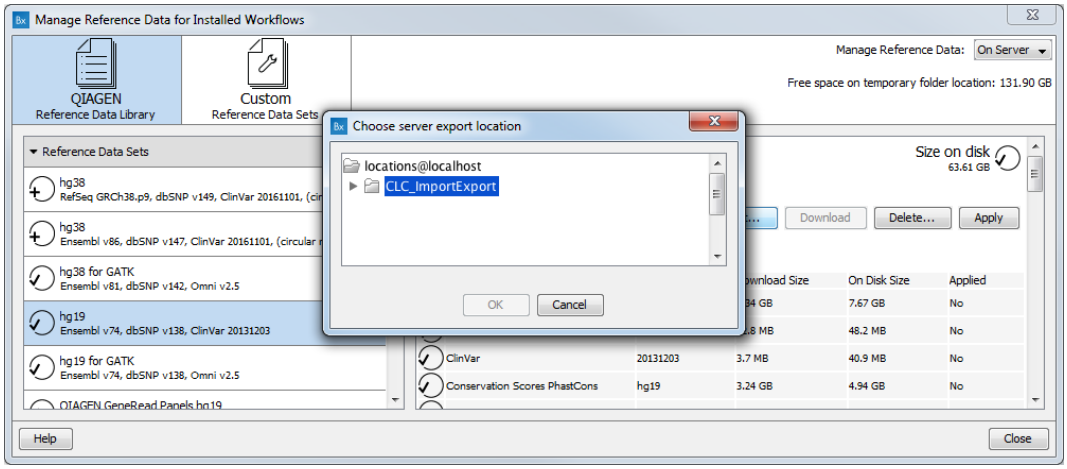


Figure 13.14: *Selecting the export location when exporting reference data in the reference data manager*

The root of an import/export location should be selected, as the tools and scripts using the exported reference data might not be able to find them otherwise. In the screenshot shown in figure 13.14, it would probably be correct to select the folder **CLC_ImportExport**.

The export is intended to be executed only once for each Reference Data Set, as Reference Data Sets do not change over time. If a Reference Data Set needs to be updated, a new version will

be created in the Reference Data Manager. This new version will be downloaded to a new version folder in the **CLC_References** folder in the persistence, and accordingly it will be exported to a new version folder on the import/export location.

The directory structure for the exported references is the same as for the "normal" references:

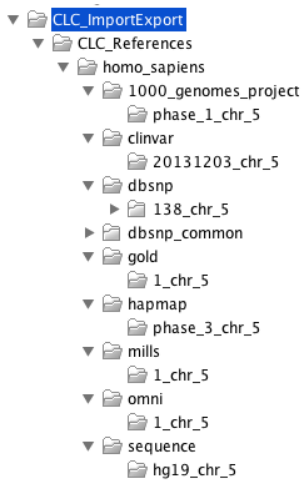


Figure 13.15: *The structure of the exported reference data.*

If the export is invoked twice for a given Reference Data Set, new files will be created in the same folder and next to the files that were exported in the first export.

For example, if the Reference Data element "Clinvar" already has been exported, then there might be a folder called `/homo_sapiens/clinvar/20131203` with the file `Clinvar_20131203.vcf` If the export is invoked again, then the folder will contain two identical files with difference names: `Clinvar_20131203.vcf` and `Clinvar_20131203.1.vcf`. The second file will not be used.

No special permissions are required to export reference data, but administrator rights are required to delete reference data. If it becomes necessary to delete exported reference data, an administrator, super user, or some user with administrator rights, must do this. Deletion of exported data has to be done through the operation system, it cannot be done through the Workbench, nor the CLC server.

13.4 Troubleshooting reference data downloads

Network connection errors can occur when downloading reference data. If this happens, you can try to resume the download from the Process tile when the network connection has been restored (see figure 13.16). Alternatively, you can simply press stop to cancel the download process and clean up any temporary data.

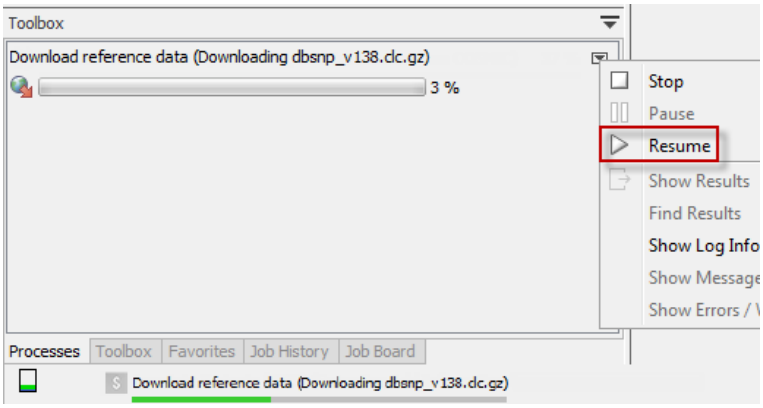


Figure 13.16: *It is possible to resume the download of data if you have encountered e.g. network connection errors.*

Chapter 14

Preparing raw data

The first thing to do after data import is to check the quality of the sequencing reads and perform the necessary trimming. This applies no matter whether you are working with Whole Genome Sequencing, Whole Exome Sequencing, Targeted Amplicon Sequencing or Whole Transcriptome Sequencing. In the toolbox there are two different ready-to-use workflows for data preparation, but the **Prepare Raw Data** ready-to-use workflow is universal and should be used for all applications.

One important part of the preparation of raw data is adapter trimming. To be able to trim off the adaptors, an adapter trim list is required. To obtain this file you will have to get in contact with the sequencing technology vendor and ask them to send this adapter trim list file to you. To learn how to create an adapter trim list, see section 21.2.3.

14.1 Prepare Overlapping Raw Data (not recommended)

The use of this workflow is not recommended any more, and **Prepare Raw Data** should be used instead.

If you wish to use this workflow anyway, launch the "Prepare Overlapping Raw Data" and select the reads that you wish to prepare for further analyses. In the wizard shown in figure 14.1, specify the trimming parameters and select the adapter trim list.

Finally, choose to **Save** the results. **Prepare Overlapping Raw Data** performs quality control and trimming of the sequencing reads. It also merges overlapping read pairs. The following outputs are generated:

- QC graphic report. The report should be checked by the user.
- QC supplementary report. The report should be checked by the user.
- Trimming report (the trimmed sequences are automatically used as input in the merging of paired reads step). The report should be checked by the user.
- Merged reads output. Use as input together with the "Not merged reads output" in the next ready-to-use workflow.
- Not merged reads output. These should be used as input together with the "Merged reads output" in the next ready-to-use workflow.

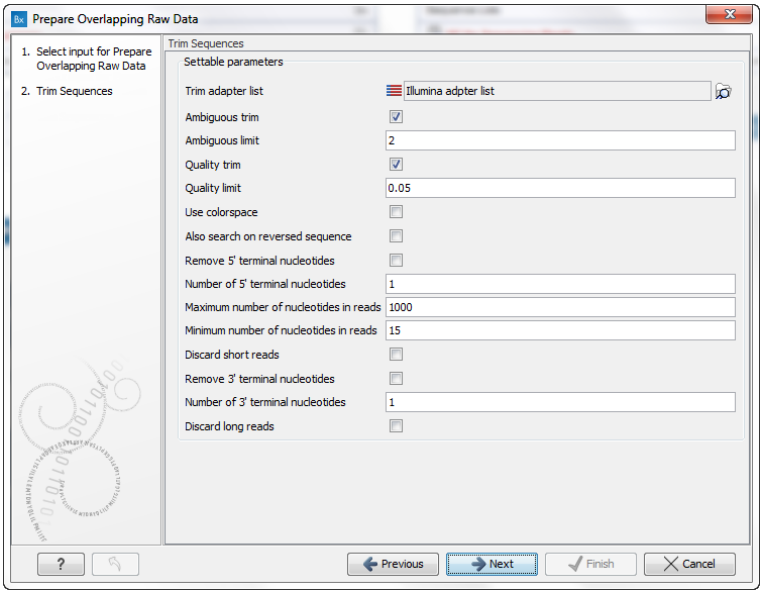


Figure 14.1: *Select your adapter trim list. You can use the default trim parameters or adjust them if necessary.*

- Broken pairs. We do not recommend to use them as input in the next ready-to-use workflow.

14.2 Prepare Raw Data (recommended)

If you have sequencing reads without overlapping pairs, you can use the "Prepare Raw Data" ready-to-use workflow for preparation of your sequences before you proceed to data analysis such as variant calling.

1. Go to the toolbox and double-click on the "Prepare Raw Data" ready-to-use workflow (figure 14.2).

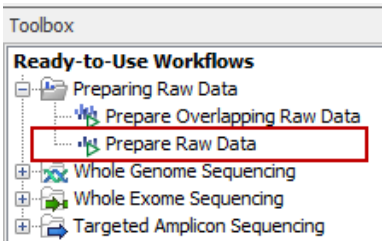


Figure 14.2: *The ready-to-use workflows are found in the toolbox.*

This will open the wizard shown in figure 14.3 where you can select the reads that you wish to prepare for further analyses.

There are three ways you can prepare your data: you can run them through the workflow one sample at the time, or you can select several samples and prepare them simultaneously, or finally you can run them in batch mode (recommended if your data are found in separate

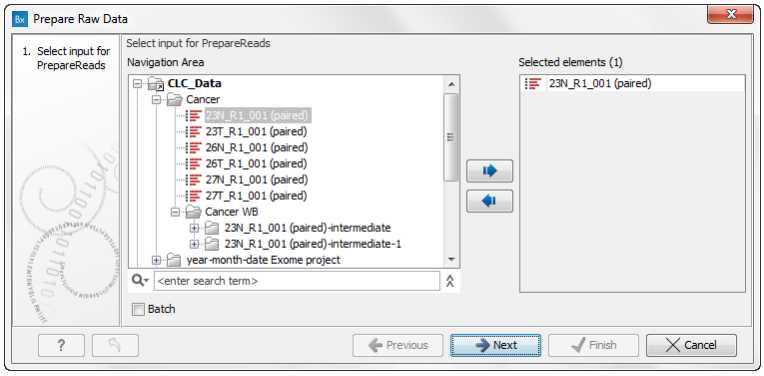


Figure 14.3: *Select the sequencing raw data that you wish to prepare before further analysis. At this step you can also choose whether you wish to prepare several reads in batch mode.*

folders). If you use batch mode, you will get an individual report for every single sample, whereas you will get one combined report for all samples if you do not run in batch mode.

To run several samples at once, select multiple samples from the left hand side list and use the small arrow pointing to the right side in the middle of the wizard to send them to "Selected elements" in the right side of the wizard. To run the samples in "Batch" mode, tick "Batch" at the bottom of the wizard as shown in figure 14.3 and select the **folder** that holds the data you wish to analyze.

2. When you have selected the sample(s) you want to prepare, click **Next**.

   As part of the data preparation, the sequences are trimmed. In the next wizard (figure 14.4) you can specify different trimming parameters and select the adapter trim list that should be used for adapter trimming by clicking on the folder icon ( ). To obtain this file you will have to get in contact with the vendor and ask them to send this adapter trim list file to you. The adapter trim list has been supplied by the vendor of the enrichment kit and sequencing machine. See section **??** for a description of how to import the adapter trim list.

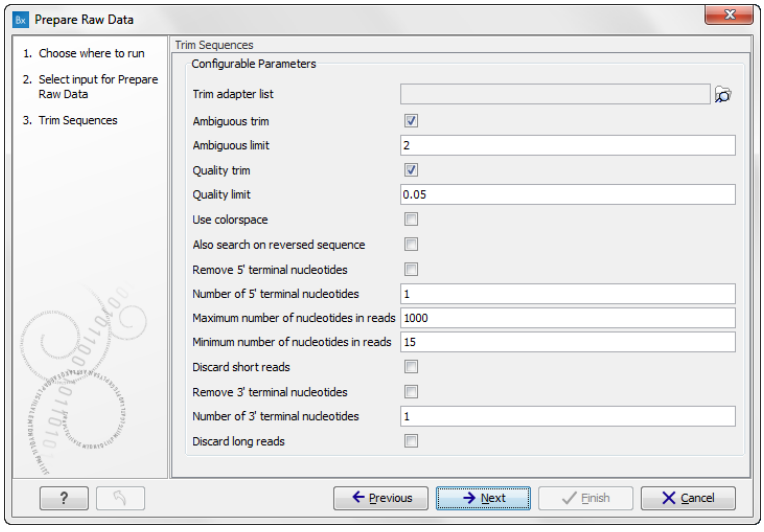


Figure 14.4: *Select your trim adapter list.*

3. Click **Next** to see the next wizard (figure 14.5).

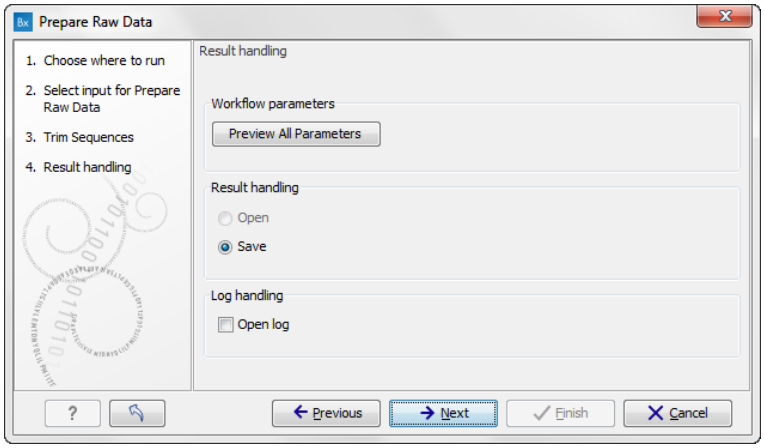


Figure 14.5: *Check the settings and save your results.*

If you click on the button labeled **Preview All Parameters** you get the chance to check the selected settings. If you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

The settings can be exported with the two buttons found at the bottom of this wizard; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

4. Click on the button labeled **OK** to go back to the previous wizard and choose **Save**.

14.2.1 Output from the Prepare Raw Data workflow

Prepare Raw Data performs quality control and trimming of the sequencing reads and generates the following outputs(figure 14.6).

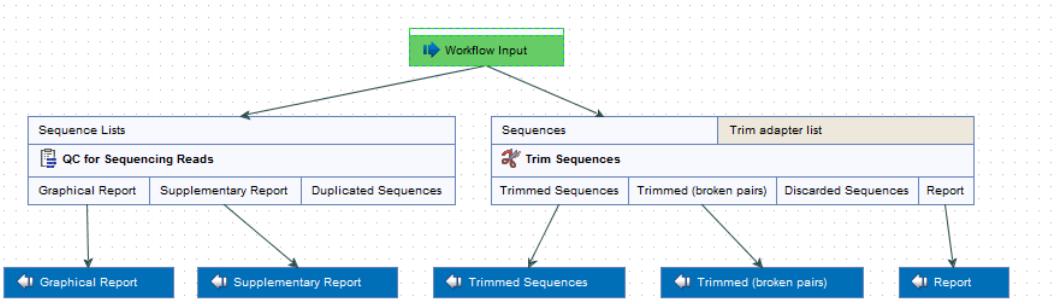


Figure 14.6: *Check the settings and save your results.*

1. QC graphic report. The report should be checked by the user.
2. QC supplementary report. The report should be checked by the user.
3. Trimming report. The report should be checked by the user.

4. Trimmed sequences output. Use as input together with the "Trimmed sequences (broken pairs) output" in the next ready-to-use workflow.

5. Trimmed sequences (broken pairs) output. We do not recommend to use as input in the next ready-to-use workflow.

14.2.2 How to check the output reports

Three different reports were generated by the workflows: a trimming report and two QC reports. All of these should be inspected in order to determine whether the quality of the sequencing reads and the trimming are acceptable. The interpretation of the reports is not always completely straightforward, but as you gain experience it becomes easier. For a detailed description of the QC reports and indication on how to interpret the different values, see section 20.2.1. For the trimming report, section 21.2.5.

If you can accept the read quality you can now proceed to the next step and use the prepared reads output as input in the next ready-to-use workflow. If the quality of your reads is poor and cannot be accepted for further analysis, the best solution to the problem is to go back to start and resequence the sample.

You are now ready to perform the actual analysis of your sequencing data.

Chapter 15

Whole genome sequencing (WGS)

Contents

The most comprehensive sequencing method is whole genome sequencing that allows for identification of genetic variations and somatic mutations across the entire human genome. This type of sequencing encompasses both chromosomal and mitochondrial DNA. The advantage of sequencing the entire genome is that not only the protein-coding regions are sequenced, but information is also provided for regulatory and non-protein-coding regions.

Eleven ready-to-use workflows are available for analysis of whole genome sequencing data (figure 15.1). The concept of the pre-installed ready-to-use workflows is that read data are used as input in one end of the workflow and in the other end of the workflow you get a track based genome browser view and a table with all the identified variants, which may or may not have been subjected to different kinds of filtering and/or annotation.

In this chapter we will discuss what the individual ready-to-use workflows can be used for and go through step by step how to run the workflows.

Note! Often you will have to prepare data with one of the two **Preparing Raw Data** workflows described in section 14 before you proceed to **Automatic analysis of sequencing data (WGS)**.

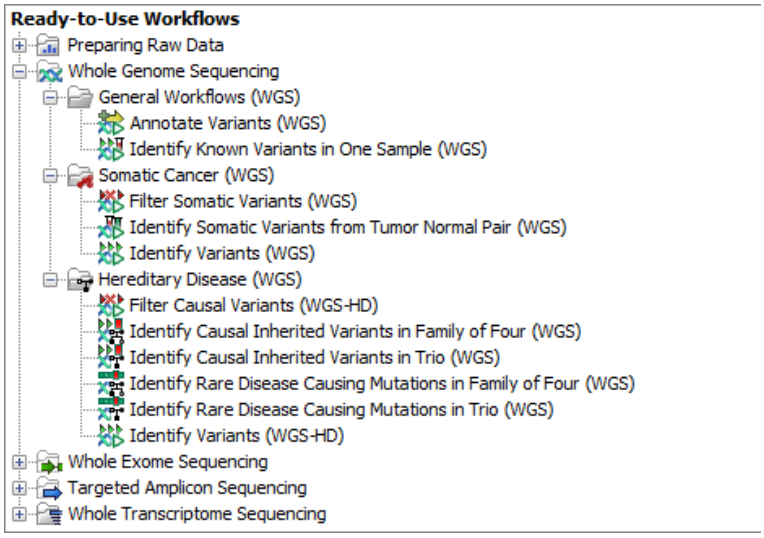


Figure 15.1: *The eleven workflows available for analyzing whole genome sequencing data.*

15.1 General Workflows (WGS)

15.1.1 Annotate Variants (WGS)

Using a variant track () (e.g. the output from the Identify Variants ready-to-use workflow) the **Annotate Variants (WGS)** ready-to-use workflow runs an internal workflow that adds the following annotations to the variant track:

- **Gene names** Adds names of genes whenever a variant is found within a known gene.
- **mRNA** Adds names of mRNA whenever a variant is found within a known transcript.
- **CDS** Adds names of CDS whenever a variant is found within a coding sequence.
- **Amino acid changes** Adds information about amino acid changes caused by the variants.
- **Information from ClinVar** Adds information about the relationships between human variations and their clinical significance.
- **Information from dbSNP** Adds information from the "Single Nucleotide Polymorphism Database", which is a general catalog of genome variation, including SNPs, multinucleotide polymorphisms (MNPs), insertions and deletions (indels), and short tandem repeats (STRs).
- **PhastCons Conservation scores** The conservation scores, in this case generated from a multiple alignment with a number of vertebrates, describe the level of nucleotide conservation in the region around each variant.

How to run the Annotate Variants (WGS) workflow

1. Go to the toolbox and select the **Annotate Variants (WGS)** workflow. In the first wizard step, select the input variant track (figure 15.2).
2. Click on the button labeled **Next**. The only parameter that should be specified by the user is which 1000 Genomes population you use (figure 15.3). This can be done using the

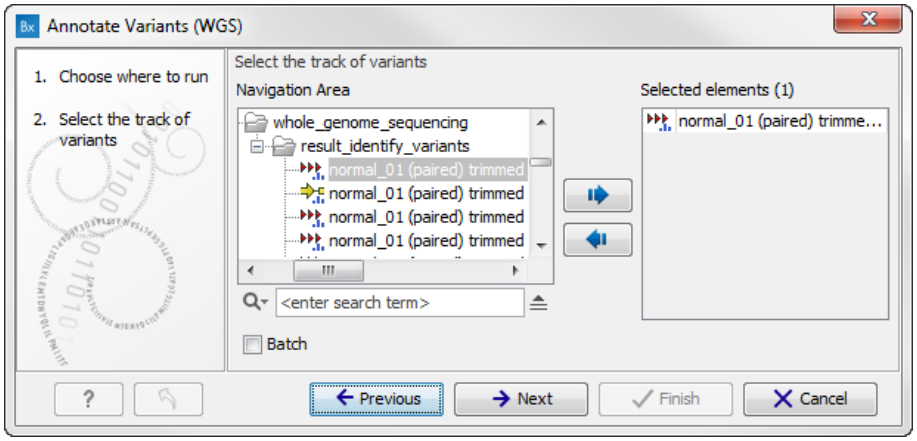


Figure 15.2: *Select the variant track to annotate.*

drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

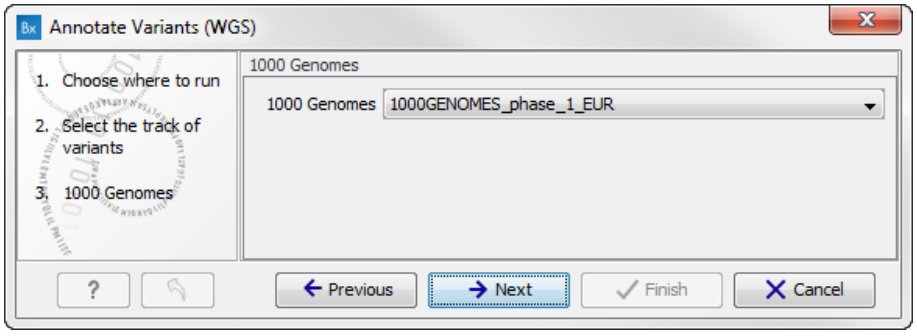


Figure 15.3: *Select the relevant 1000 Genomes population(s).*

3. Click on the button labeled **Next** to go to the last wizard step (figure 15.4).

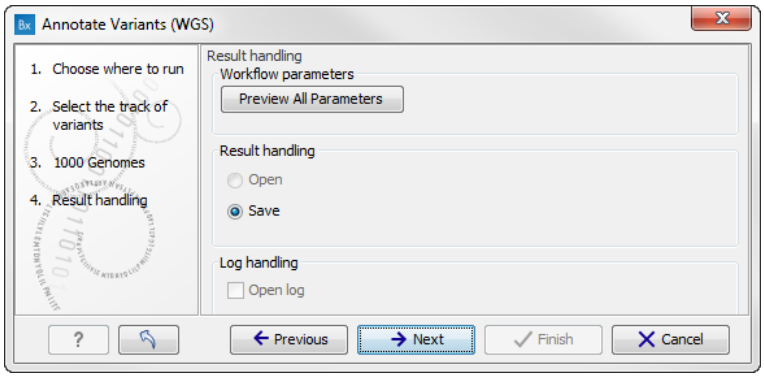


Figure 15.4: *Check the settings and save your results.*

In this wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

4. Choose to **Save** your results and click on the button labeled **Finish**.

Output from the Annotate Variants (WGS) workflow

Two types of output are generated:

1. **Annotated Variants** ( ) Annotation track showing the variants. Hold the mouse over one of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant.

2. An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.

3. **Genome Browser View Annotated Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in dbSNP, ClinVar, 1000 Genomes, and PhastCons conservation scores (see figure 15.5).

Figure 15.5: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list) containing individual tracks for all added annotations.*

Note! Please be aware, that if you delete the annotated variant track, this track will also disappear from the genome browser view.

It is possible to add tracks to the Genome Browser View such as mapped sequencing reads as well as other tracks. This can be done by dragging the track directly from the **Navigation Area** to the Genome Browser View.

If you double-click on the name of the annotated variant track in the left hand side of the Genome Browser View, a table that includes all variants and the added information/annotations will open (see figure 15.6). The table and the Genome Browser View are linked; if you click on an entry in the table, this particular position in the genome will automatically be brought into focus in the Genome Browser View.

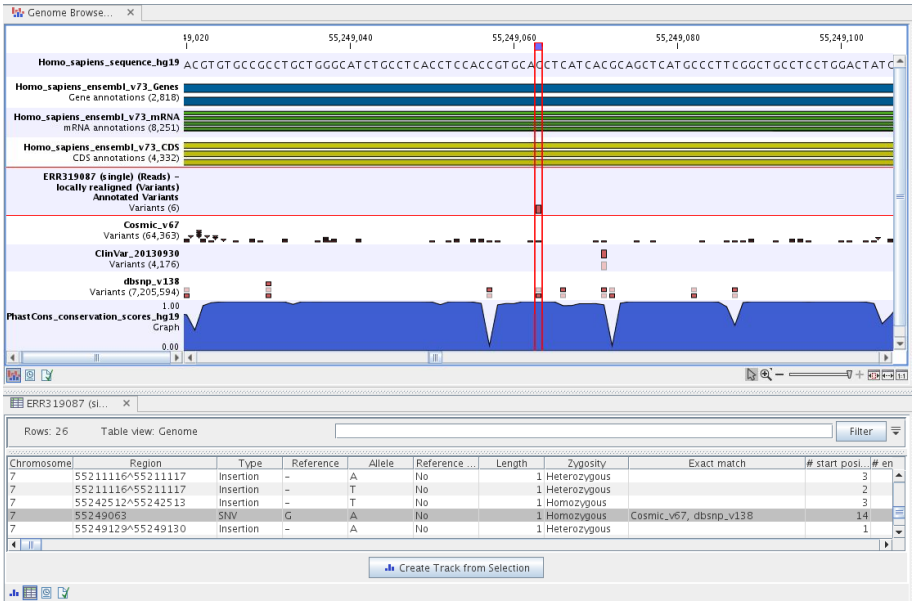


Figure 15.6: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list). The information is also available in table view. Click on the small table icon to open the table view. If you hold down the "Ctrl" key while clicking on the table icon, you will open a split view showing both the genome browser view and the table view.*

You may be met with a warning as shown in figure 15.7. This is simply a warning telling you that it may take some time to create the table if you are working with tracks containing large amounts of annotations. Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

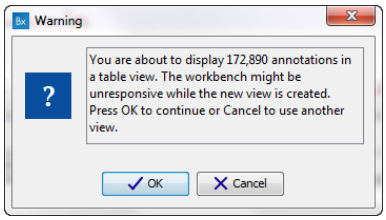


Figure 15.7: *Warning that appears when you work with tracks containing many annotations.*

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar database, can be

prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level between different vertebrates or mammals in the region containing the variant can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) are prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** () | **Create Filter Criteria** ()

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

15.1.2 Identify Known Variants in One Sample (WGS)

The **Identify Known Variants in One Sample (WGS)** ready-to-use workflow is a combined data analysis and interpretation ready-to-use workflow.

It should be used to identify known variants, specified by the user (e.g. known breast cancer associated variants), for their presence or absence in a sample.

Please note that the ready-to-use workflow will not identify new variants.

The **Identify Known Variants in One Sample (WGS)** ready-to-use workflow maps the sequencing reads to a human genome sequence and does a local realignment of the mapped reads to improve the subsequent variant detection. In the next step, only variants specified by the user are identified and annotated in the newly generated read mapping.

Import your known variants

To make an import into the *Biomedical Genomics Workbench*, you should have your variants in GVF format (http://www.sequenceontology.org/resources/gvf.html) or VCF format (http://ga4gh.org/#/fileformats-team).

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

How to run the Identify Known Variants in One Sample (WGS) workflow

1. Go to the toolbox and double-click on

Toolbox | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **General Workflows (WGS)** | **Identify Known Variants from One Sample (WGS)** ( )

2. This will open the wizard step shown in figure 15.8 where you can select the reads of the sample that should be tested for presence or absence of your known variants.

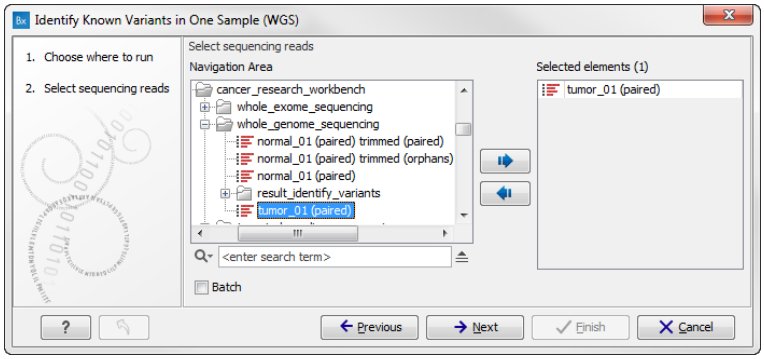


Figure 15.8: *Select the sequencing reads from the sample you would like to test for your known variants.*

If several samples from different folders should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" and specifying the folders that hold the data you wish to analyse.

Click on the button labeled **Next**.

3. In the next wizard step, select the file containing the known variants you want to identify in the read mapping (figure 15.9).

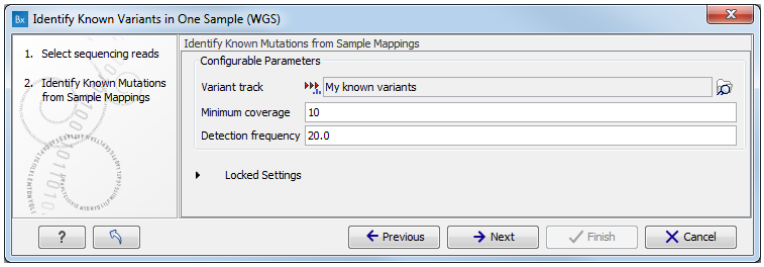


Figure 15.9: *Specify the track with the known variants that should be identified.*

The parameters that can be set are:

- **Minimum coverage** The minimum number of reads that covers the position of the variant, which is required to set "Sufficient Coverage" to YES.
- **Detection frequency** The minimum allele frequency that is required to annotate a variant as being present in the sample. The same threshold will also be used to determine if a variant is homozygous or heterozygous. In case the most frequent alternative allele at the position of the considered variant has a frequency of less than this value, the zygosity of the considered variant will be reported as being homozygous.

The parameter "Detection Frequency" will be used in the calculation twice. First, it will report in the result if a variant has been detected (observed frequency > specified frequency) or not (observed frequency <= specified frequency). Moreover, it will determine if a variant

should be labeled as heterozygous (frequency of another allele identified at a position of a variant in the alignment > specified frequency) or homozygous (frequency of all other alleles identified at a position of a variant in the alignment < specified frequency).

Click on the button labeled **Next**.

4. In the last wizard step (figure 15.10) you can check the selected settings by clicking on the button labeled **Preview All Parameters**.

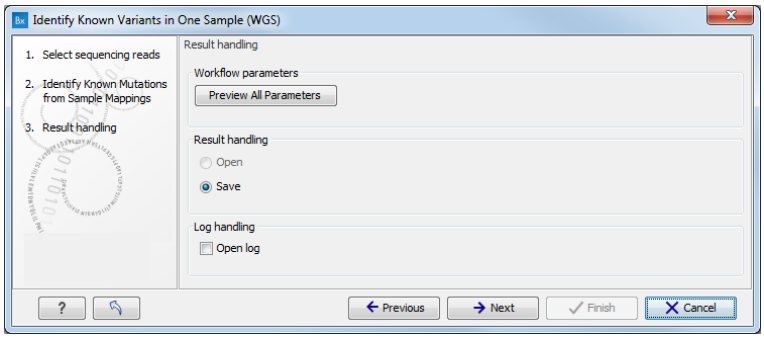


Figure 15.10: *Check the settings and save your results.*

At the bottom of this wizard there are two buttons regarding export functions: one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

5. Click on the button labeled **OK** to go back to the previous dialog box and choose **Save**.

   Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Known Variants in One Sample (WGS) workflow

The **Identify Known Variants in One Sample (WGS)** tool produces four different output types.

1. **Read Mapping Report** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads.

2. **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

3. **Variants Detected in Detail** ( ) Annotation track showing the known variants. Like the "Overview Variants Detected" table, this table provides information about the known variants. Four columns starting with the sample name and followed by "Read Mapping coverage", "Read Mapping detection", "Read Mapping frequency", and "Read Mapping zygosity" provides the overview of whether or not the known variants have been detected in the sequencing reads, as well as detailed information about the Most Frequent Alternative Allele (labeled MFAA).

4. **Genome Browser View Identify Known Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, target regions coverage, the mapped reads, the overview of the detected variants, and the variants detected in detail.

It is a good idea to start looking at the mapping report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Please also check that at least 90% of the reads are mapped to the human reference sequence.

When this has been done you can open the Genome Browser View file (see 15.11).

The Genome Browser View includes the overview track of known variants and the detailed result track in the context to the human reference sequence, genes, transcripts, coding regions, targeted regions, and mapped sequencing reads.

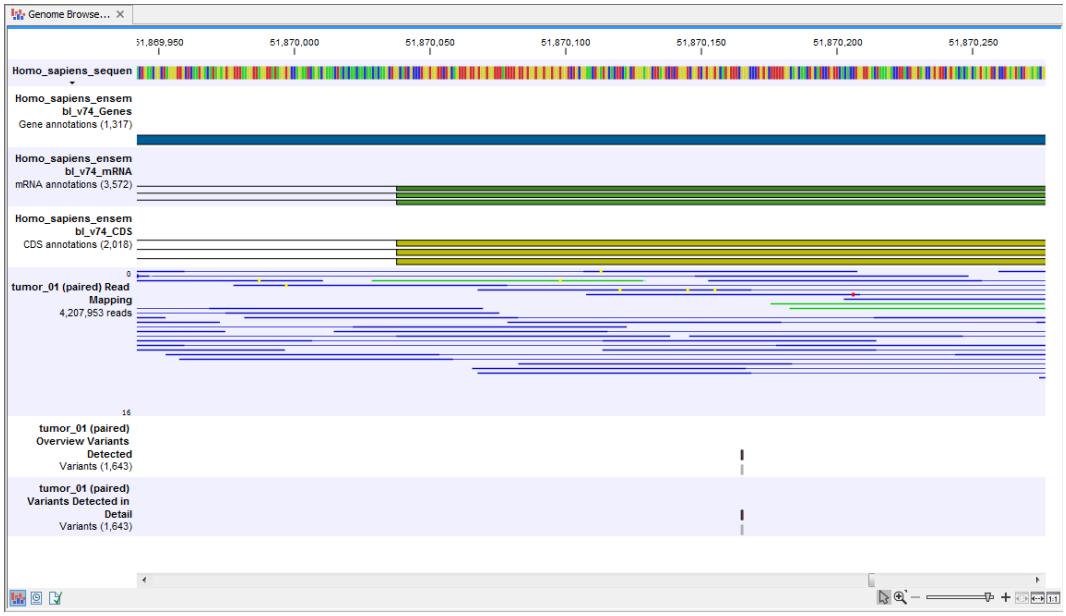


Figure 15.11: *Genome Browser View that allows inspection of the identified variants in the context of the human genome and external databases.*

Finally, a track with conservation scores has been added to be able to see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant.

By double clicking on one of the annotated variant tracks in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see 15.12).

Note We do not recommend that any of the produced files are deleted individually as some of them are linked to other outputs. Please always delete all of them at the same time.

15.2 Somatic Cancer (WGS)

15.2.1 Filter Somatic Variants (WGS)

If you are analyzing a list of variants that have been detected in a tumor or blood sample where no control sample is available from the same patient, you can use the **Filter Somatic**

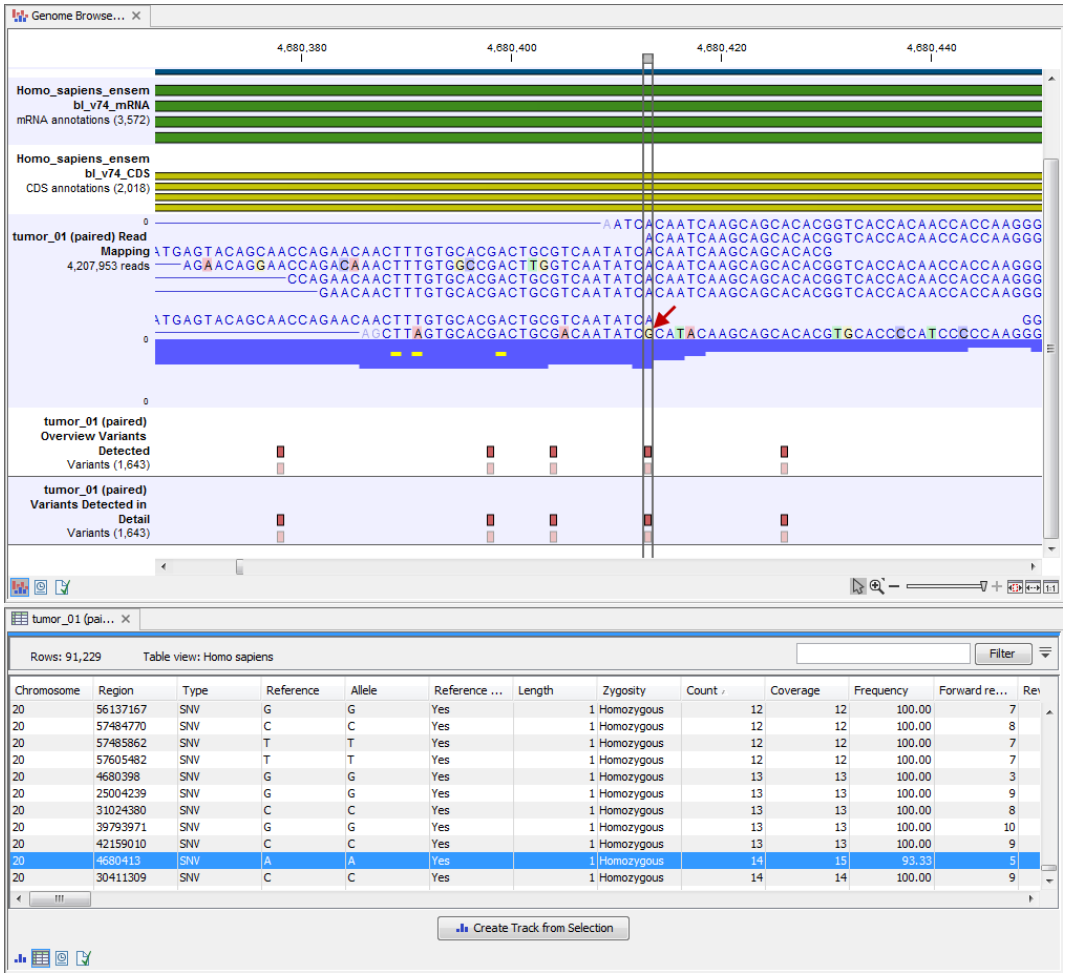


Figure 15.12: *Genome Browser View with an open overview variant track with information about if the variant has been detected or not, the identified zygosity, if the coverage was sufficient at this position and the observed allele frequency.*

Variants (WGS) ready-to-use workflow to identify potential somatic variants. The purpose of this ready-to-use workflow is to use publicly available (or your own) databases, with common variants in a population, to extract potential somatic variants whenever no control/normal sample from the same patient is available.

The **Filter Somatic Variants (WGS)** ready-to-use workflow accepts variant tracks () (e.g. the output from the Identify Variants ready-to-use workflow) as input. Variants that are identical to the human reference sequence are first filtered away and then variants found in the Common dbSNP, 1000 Genomes Project, and HapMap databases are deleted. Variants in those databases are assumed to not contain relevant somatic variants.

Please note that this tool will likely also remove inherited cancer variants that are present at a low percentage in a population.

Next, the remaining somatic variants are annotated with gene names, amino acid changes, conservation scores and information from ClinVar (known variants with medical impact) and dbSNP (all known variants).

How to run the Filter Somatic Variants (WGS) workflow

To run the **Filter Somatic Variants (WGS)** tool, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Genome Sequencing** () | **Somatic Cancer** () | **Filter Somatic Variants** ()

1. Double-click on the **Filter Somatic Variants (WGS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Next, you will be asked to select the variant track you would like to use for filtering somatic variants. The panel in the left side of the wizard shows the kind of input that should be provided (figure 15.13). Select by double-clicking on the reads file name or clicking once on the file and then clicking on the arrow pointing to the right side in the middle of the wizard.

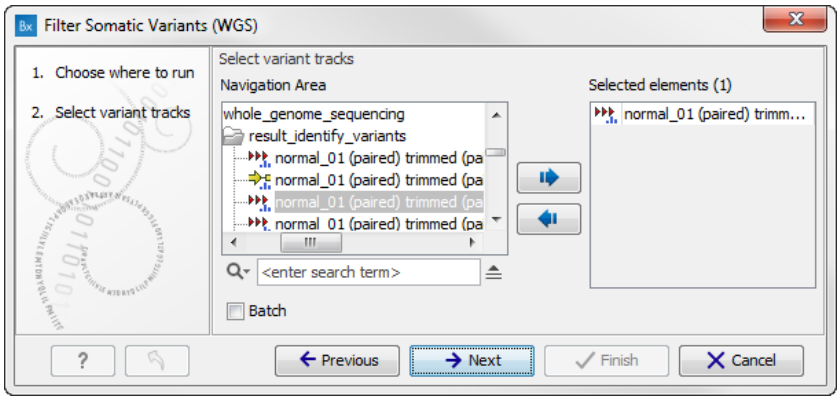


Figure 15.13: *Select the variant track from which you would like to filter somatic variants.*

Click on the button labeled **Next**.

3. In the next step you will be asked to specify which of the 1000 Genomes populations that should be used for annotation (figure 15.14).

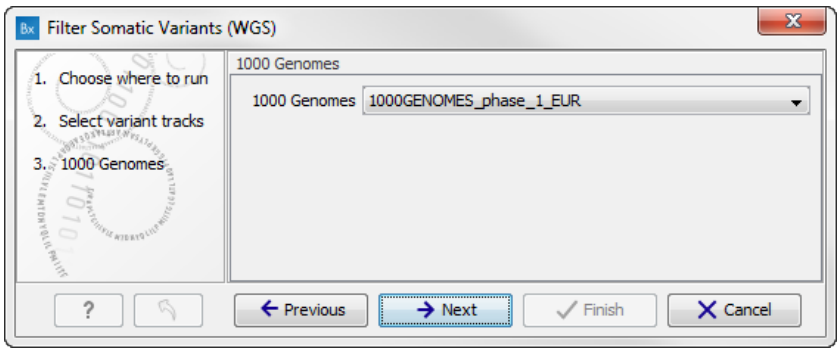


Figure 15.14: *Specify which 1000 Genomes population to use for annotation.*

Click on the button labeled **Next**.

4. The next wizard step will once again allow you to specify the 1000 Genomes population that should be used, this time for filtering out variants found in the 1000 Genomes project (figure 15.15).

   Click on the button labeled **Next**.

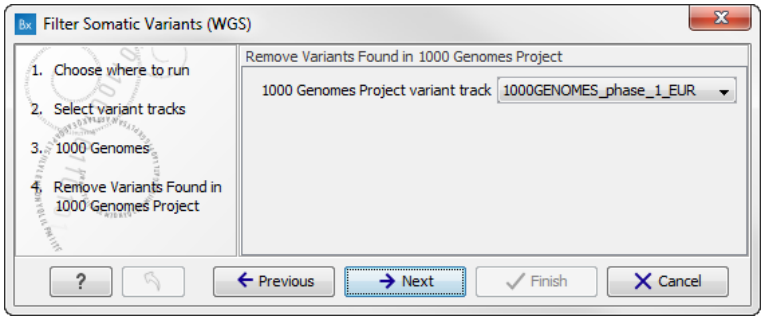


Figure 15.15: *Specify which 1000 Genomes population to use for filtering out known variants.*

5. The next wizard step (figure 15.16) concerns removal of variants found in the HapMap database. Select the population you would like to use from the drop-down list. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

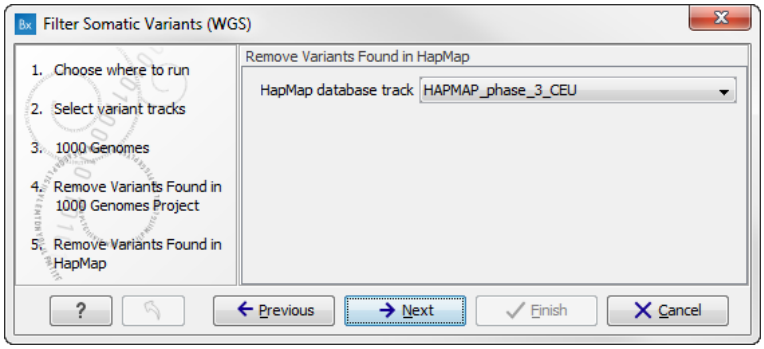


Figure 15.16: *Specify which HapMap population to use for filtering out known variants.*

6. Click on the button labeled **Next** to go to the last wizard step (shown in figure 15.17).

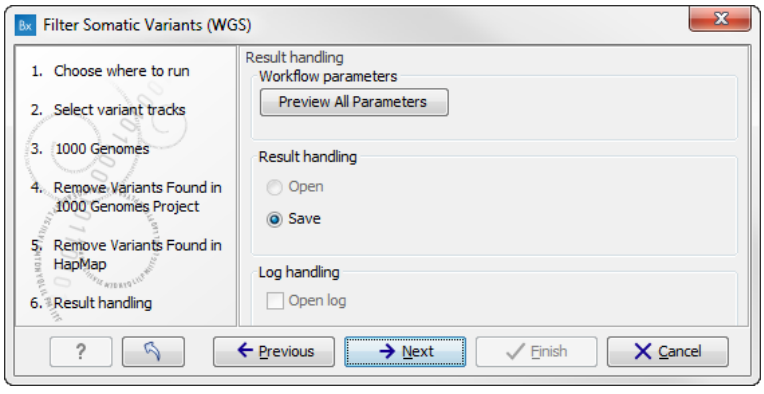


Figure 15.17: *Check the selected parametes by pressing "Preview All Parameters".*

Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Filter Somatic Variants (WGS) workflow

Two types of output are generated:

1. **Somatic Candidate Variants** Track that holds the variant data. This track is also included in the Genome Browser View. If you hold down the Ctrl key (Cmd on Mac) while clicking on the table icon in the lower left side of the **View Area**, you can open the table view in split view. The table and the variant track are linked together, and when you click on a row in the table, the track view will automatically bring this position into focus.
2. **Genome Browser View Filter Somatic Variants** A collection of tracks presented together. Shows the somatic candidate variants together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar, 1000 Genomes, and the PhastCons conservation scores (see figure 15.18).

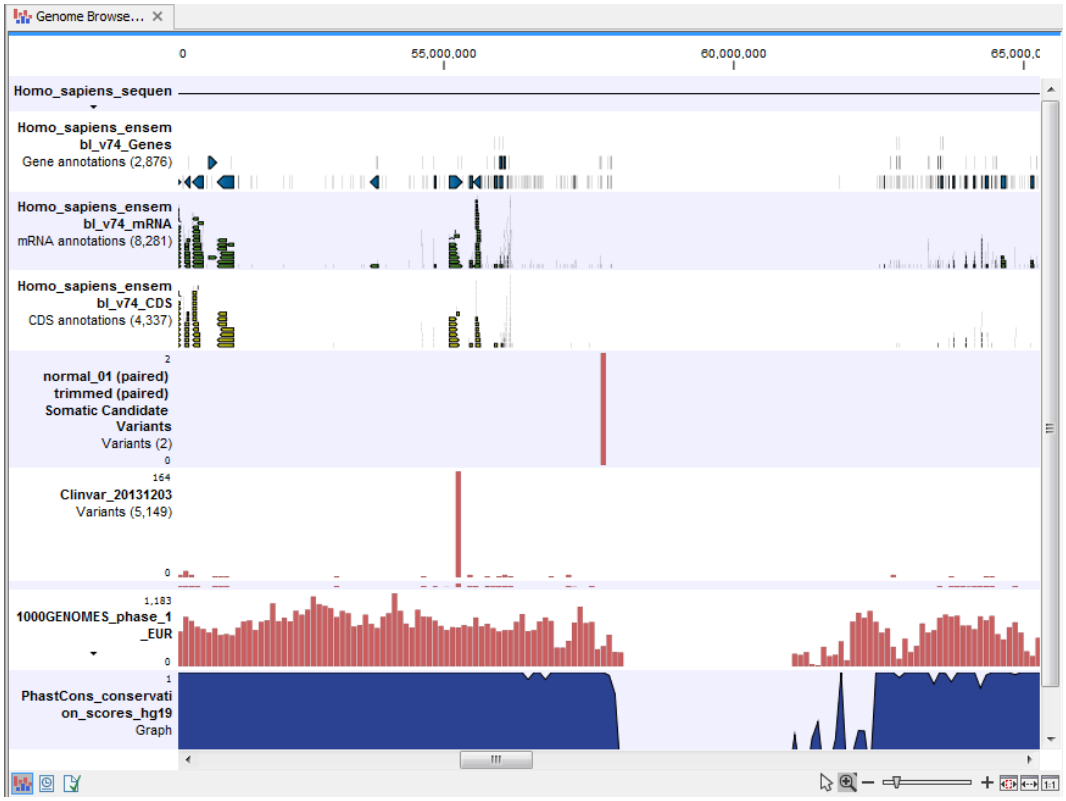


Figure 15.18: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

The track with the conservation scores allows you to see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant. Mapped sequencing reads as well as other tracks can be easily added to the Genome Browser View.

If you click on the annotated variant track in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations. This is shown in figure 15.19.

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar

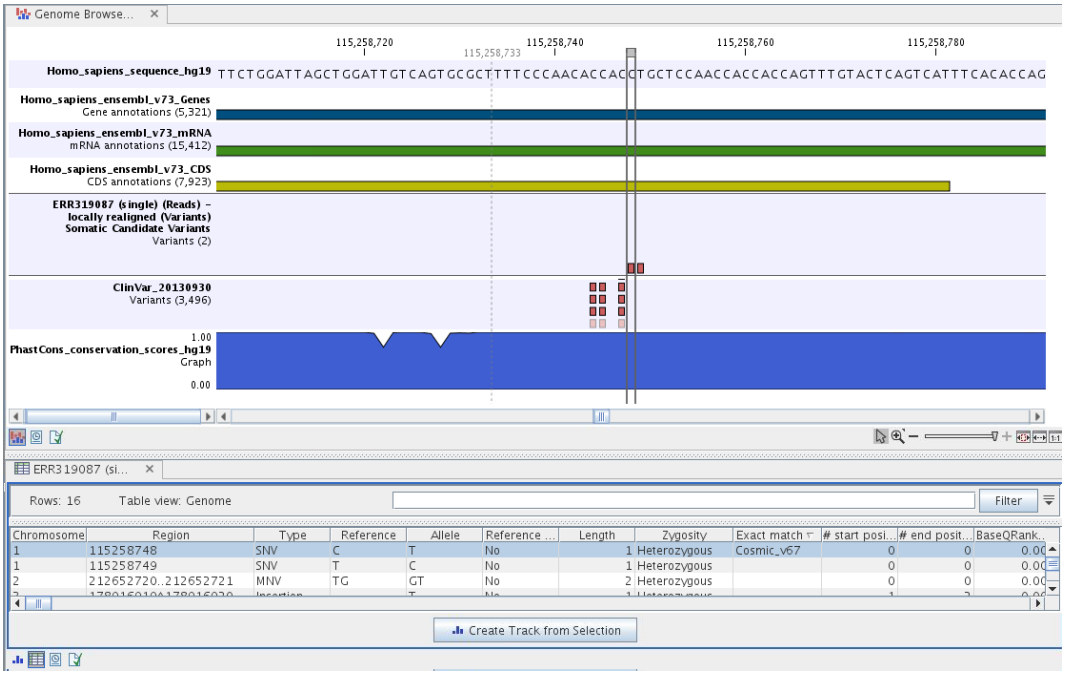


Figure 15.19: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

database) can easily be identified. Further, variants not found in the ClinVar databases, can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level, between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** () | **Create Filter Criteria** ()

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

15.2.2 Identify Somatic Variants from Tumor Normal Pair (WGS)

The **Identify Somatic Variants from Tumor Normal Pair (WGS)** ready-to-use workflow can be used to identify potential somatic variants in a tumor sample when you also have a normal/control sample from the same patient.

When running the **Identify Somatic Variants from Tumor Normal Pair (WGS)** the reads are mapped and the variants identified. An internal workflow removes germline variants that are found in the mapped reads of the normal/control sample and variants outside the target region are removed as they are likely to be false positives due to non-specific mapping of sequencing reads. Next, remaining variants are annotated with gene names, amino acid changes, conservation scores and information from relevant databases like ClinVar (variants with clinically relevant association). Finally, information from dbSNP is added to see which of the detected variants have been observed before and which are completely new.

How to run the Identify Somatic Variants from Tumor Normal Pair (WGS) workflow

To run the **Identify Somatic Variants from Tumor Normal Pair (WGS)** tool, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **Somatic Cancer** ( ) | **Identify Somatic Variants from Tumor Normal Pair (WGS)** ( )

1. Go to the toolbox and double-click on the **Identify Somatic Variants from Tumor Normal Pair (WGS)** ready-to-use workflow. This will open the wizard shown in figure 15.20 where you can select the tumor sample reads.

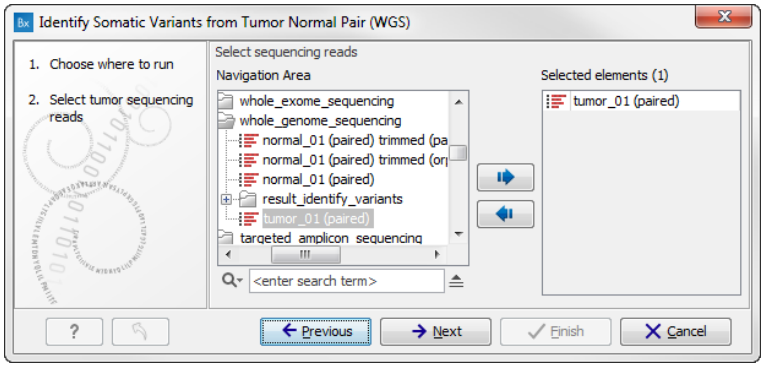


Figure 15.20: *Select the tumor sample reads.*

When you have selected the tumor sample reads click on the button labeled **Next**.

2. In the next wizard step (figure 15.21), please specify the normal sample reads.

3. In the next wizard step you can adjust the settings used for variant detection (figure 15.22). For a description of the different parameters that can be adjusted, see section 22.14. If you click on "Locked Settings", you will be able to see all parameters used for variant detection in the ready-to-use workflow.

   In this dialog, you have to specify the parameters for the variant detection. For a description of the different parameters that can be adjusted, see section 22.14. If you click on "Locked Settings", you will be able to see all parameters used for variant detection in the ready-to-use workflow.

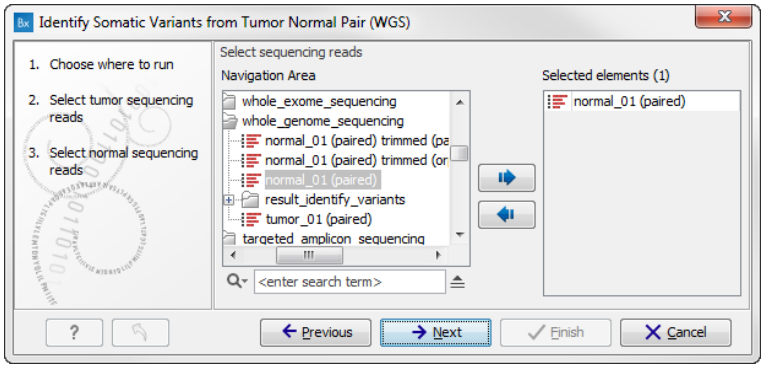


Figure 15.21: *Select the normal sample reads.*

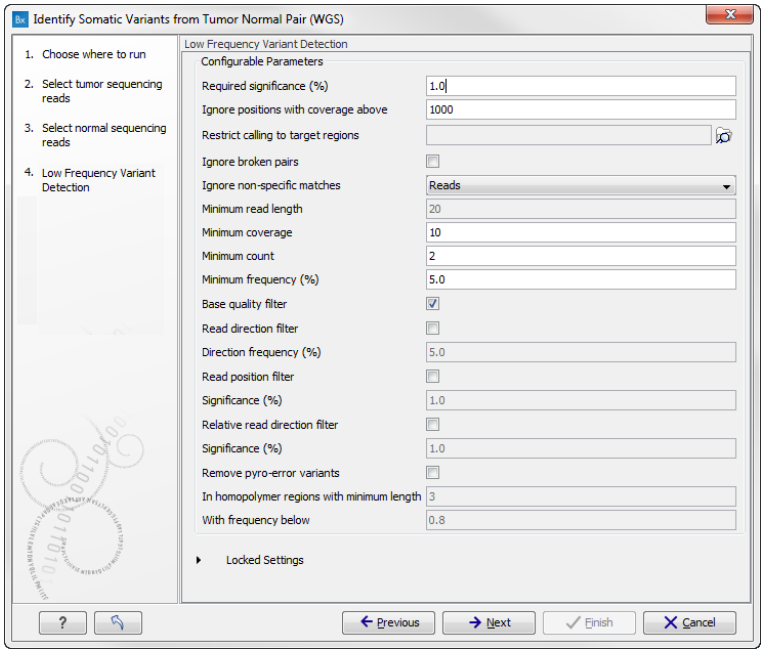


Figure 15.22: *Specify the settings for the variant detection.*

4. Click on the button labeled **Next** to go to the step where you can adjust the settings for removal of germline variants (figure 15.23).

   Click on the button labeled **Next**.

5. In the next wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters** (figure 15.24).

   In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

6. Click on the button labeled **OK** to go back to the previous wizard step and choose **Save**.

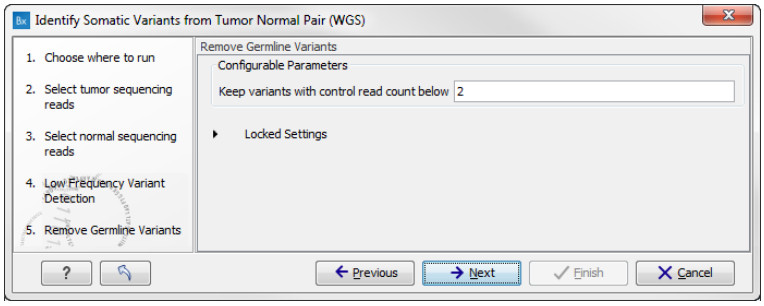


Figure 15.23: *Specify setting for removal of germline variants.*

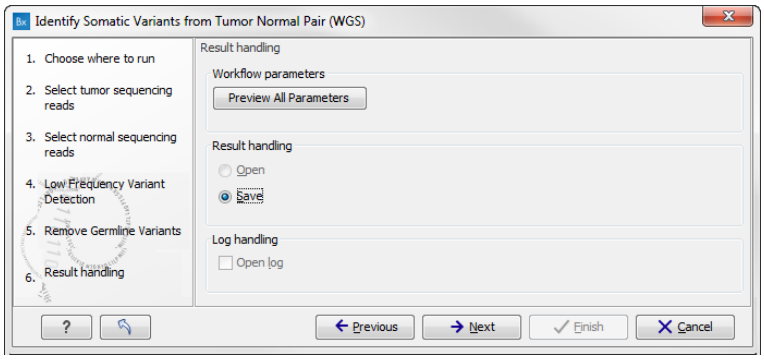


Figure 15.24: *Check the parameters and save the results.*

Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Somatic Variants from Tumor Normal Pair (WGS) workflow

Seven different outputs are generated:

1. **Read Mapping Tumor** ( ) The mapped sequencing reads for the tumor sample. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

2. **Read Mapping Normal** ( ) The mapped sequencing reads for the normal sample. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously.

3. **Mapping Report Tumor** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from the tumor sample.

4. **Mapping Report Normal** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from the normal sample.

5. **Amino Acids Changes** Track that shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.

6. **Annotated Somatic Variants** ( ) A variant track holding the identified and annotated somatic variants. The variants can be shown in track format or in table format. When

holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

7. **Genome Browser View Tumor Normal Comparison** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads for both normal and tumor, the annotated somatic variants, information from the ClinVar database, and finally a track showing the conservation score (see figure 15.25).

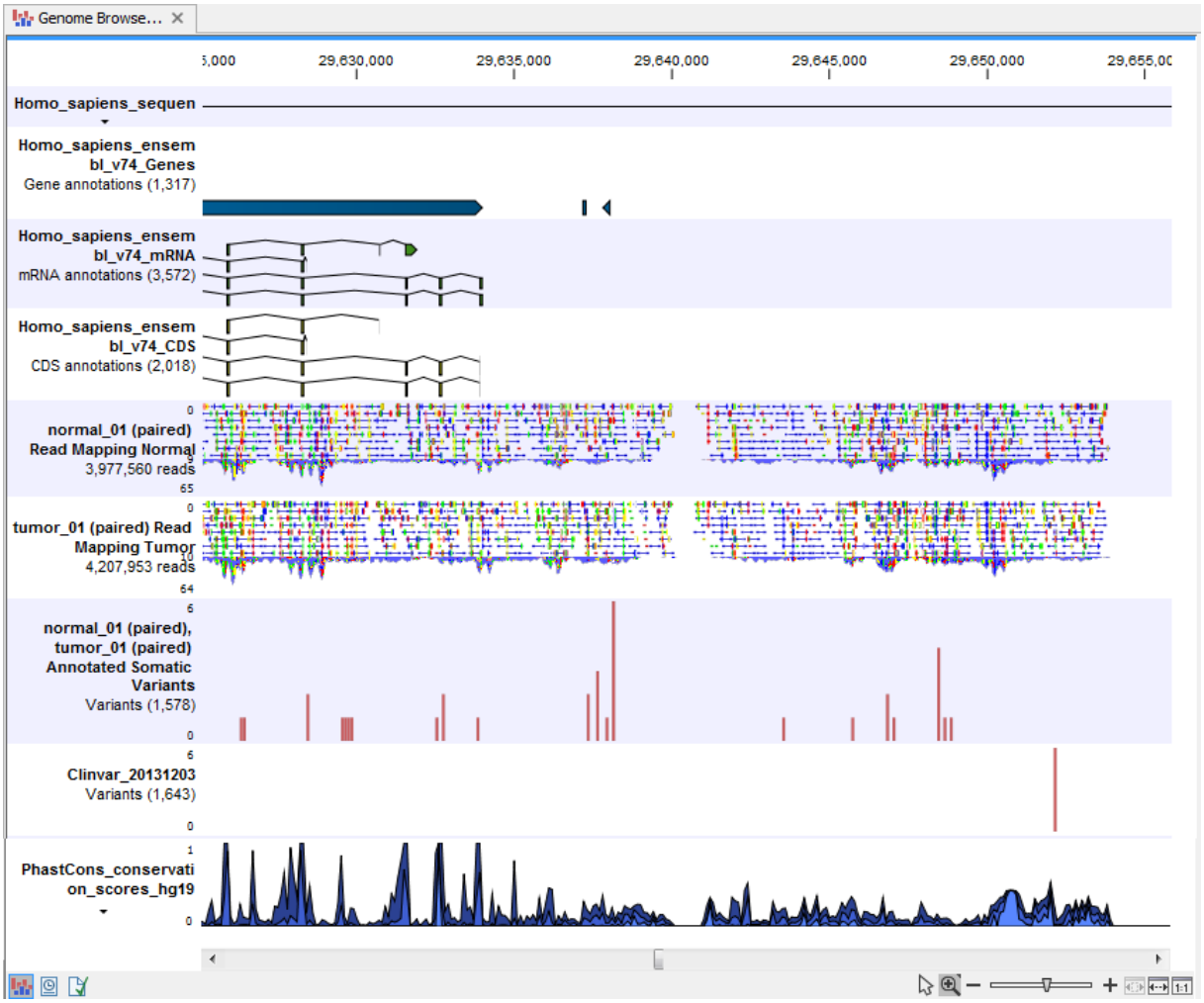


Figure 15.25: *The Genome Browser View presents all the different data tracks together and makes it easy to compare different tracks.*

15.2.3 Identify Variants (WGS)

The **Identify Variants (WGS)** tool takes sequencing reads as input and returns identified variants in a Genome Browser View.

The tool runs an internal workflow that first maps the sequencing reads to the human reference sequence. Next, it runs a local realignment that is used to improve the variant detection

that comes after the local realignment. Two different variant callers are used; the **Low Frequency Variant Detection** tool that is used to call small insertions, deletions, SNVs, MNV, and replacements, and the "InDel and Structural Variants" caller that calls larger insertions, deletions, translocations, and replacements. By the end of the variant detection, variants that have been detected by the **Low Frequency Variant Detection** caller with an average base quality smaller than 20 are filtered away.

A detailed mapping report is created to inspect the overall coverage and mapping specificity in the targeted regions.

How to run the Identify Variants (WGS) workflow

To run the **Identify Variants (WGS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Genome Sequencing** () | **Somatic Cancer** () | **Identify Variants (WGS)** ()

1. Select the sequencing reads from the sample that should be analyzed (figure 15.26).

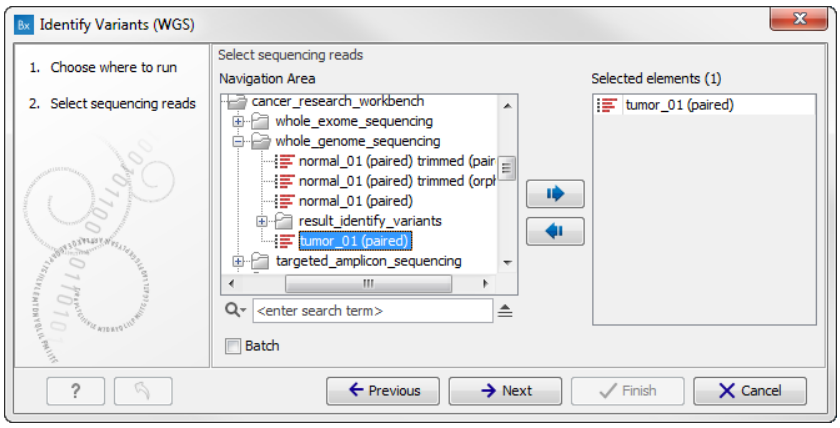


Figure 15.26: *Please select all sequencing reads from the sample to be analyzed.*

   Select all sequencing reads from your sample. If several samples should be analyzed, the tool has to be run in batch mode. To do this, tick "Batch" at the bottom of the wizard and select the **folder** that holds the data you wish to analyze.

   If you have your sequencing data in separate folders, you should choose to run the analysis in batch mode.

   When you have selected the sample(s) that you want to prepare, click on the button labeled **Next**.

2. In the next wizard step (figure 15.27) you can specify the parameters for variant detection.

3. Click on the button labeled **Next**. This will take you to the next wizard step (figure 15.28).

   In this wizard you can check the selected settings by clicking on the button labeled **Preview All Parameters**.

   In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this wizard there are two buttons regarding export

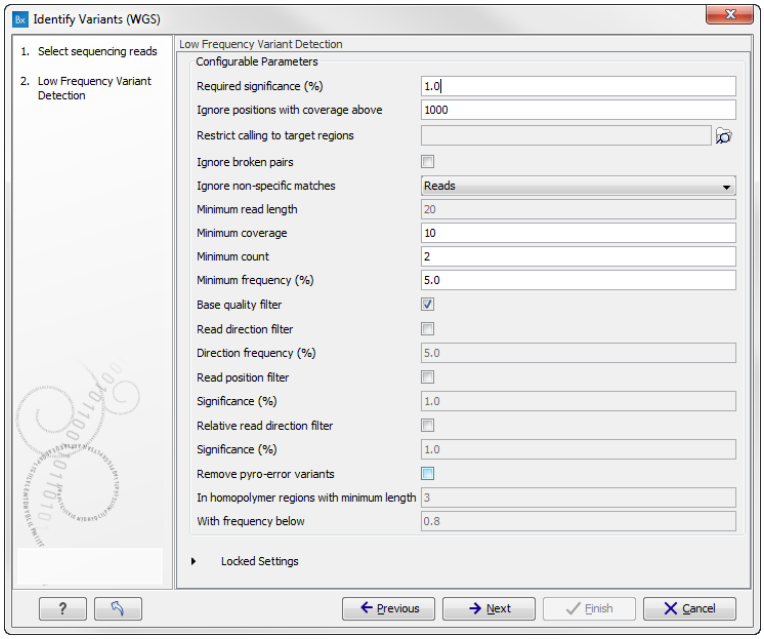


Figure 15.27: *The next thing to do is to specify the parameters that should be used to detect variants.*

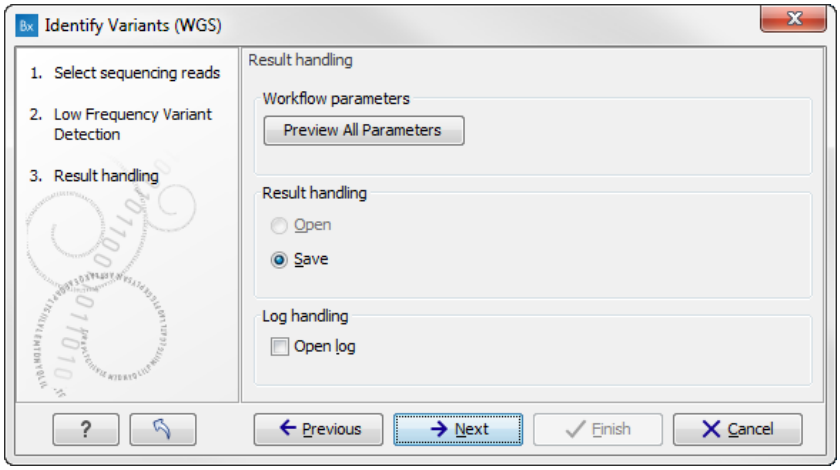


Figure 15.28: *Check the settings and save your results.*

functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

4. Click on the button labeled **OK** to go back to the previous wizard and choose **Save**.

   Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Variants (WGS) workflow

The **Identify Variants (WGS)** tool produces six different types of output:

1. **Structural Variants** ( ) Variant track showing the structural variants; insertions, deletions, replacements. Hold the mouse over one of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant. The structural variants can also be viewed in table format by switching to the table view. This is done by pressing the table icon found in the lower left corner of the **View Area**.

2. **Structural Variant Report** ( ) The report consists of a number of tables and graphs that in different ways provide information about the structural variants.

3. **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

4. **Read Mapping Report** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads.

5. **Structural Variants** ( ) A variant track holding the identified variants. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

6. **Genome Browser View Identify Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads, the identified variants, and the structural variants (see figure 15.5).

Before looking at the identified variants, we recommend that you first take a look at the mapping report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Furthermore, please check that at least 90% of the reads map to the human reference sequence. In case of a targeted experiment, please also check that the majority of reads map to the targeted region.

Next, open the Genome Browser View file (see figure 15.29).

The Genome Browser View lists the track of the identified variants in context to the human reference sequence, genes, transcripts, coding regions, and mapped sequencing reads.

By double-clicking on the indsel variant track in the Genome Browser View, a table will be shown that lists all identified larger insertions and deletions (see figure 15.30).

In case you would like to change the reference sequence used for read mapping or the human genes, please use the "Data Management" (see section 13.1).

15.3 Hereditary Disease (WGS)

15.3.1 Filter Causal Variants (WGS-HD)

If you are analyzing a list of variants, you can use the **Filter Causal Variants (WGS-HD)** ready-to-use workflow to remove variants that are outside the target region, as well as common variants present in publicly available databases. The workflow will annotate the remaining variants with gene names, conservation scores, and information from relevant databases.

The **Filter Causal Variants (WGS-HD)** ready-to-use workflow accepts variants tracks files.

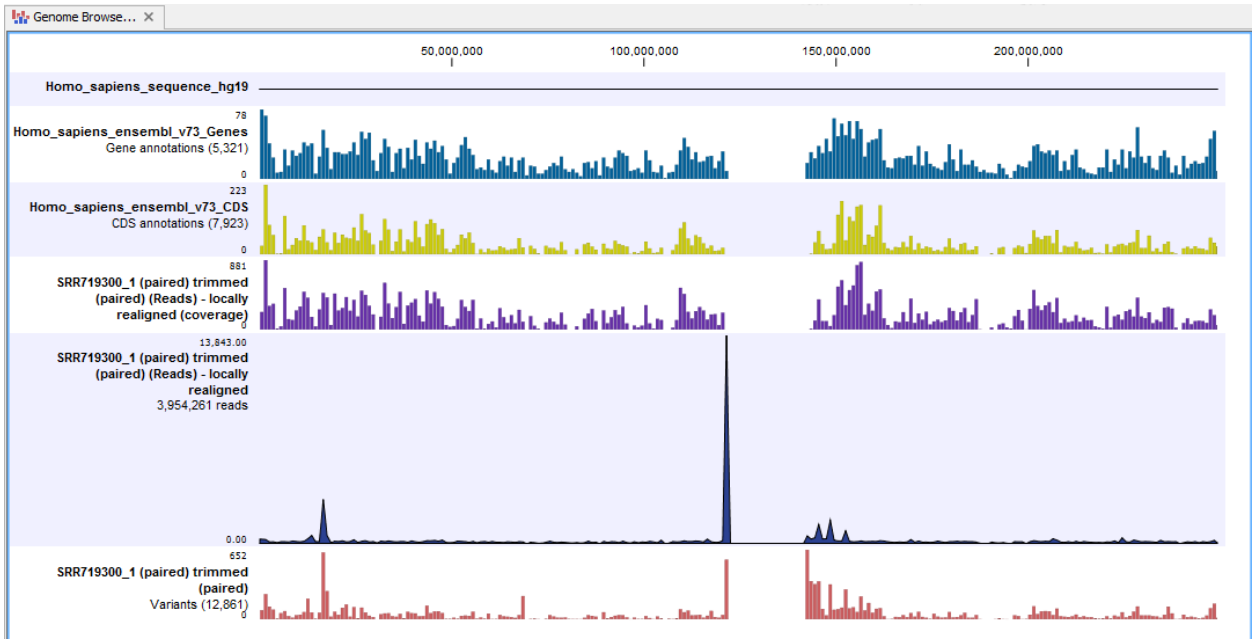


Figure 15.29: *The Genome Browser View allows easy inspection of the identified smaller variants, larger insertions and deletions, and structural variants in the context of the human genome.*

How to run the Filter Causal Variants (WGS-HD) workflow

To run the **Filter Causal Variants (WGS-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **Hereditary Disease** ( ) | **Filter Causal Variants (WGS-HD)** ( )

1. Double-click on the **Filter Causal Variants (WGS-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the **variant track** you want to use for filtering causal variants (figure 15.31).

   The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the variant track name or click once on the file and then click on the arrow pointing to the right side in the middle of the wizard.
3. Specify which of the **1000 Genomes populations** that should be used for **annotation** (figure 15.32).
4. Specify the **1000 Genomes population** that should be used for **filtering out** variants found in the 1000 Genomes project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).
5. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 17.50).

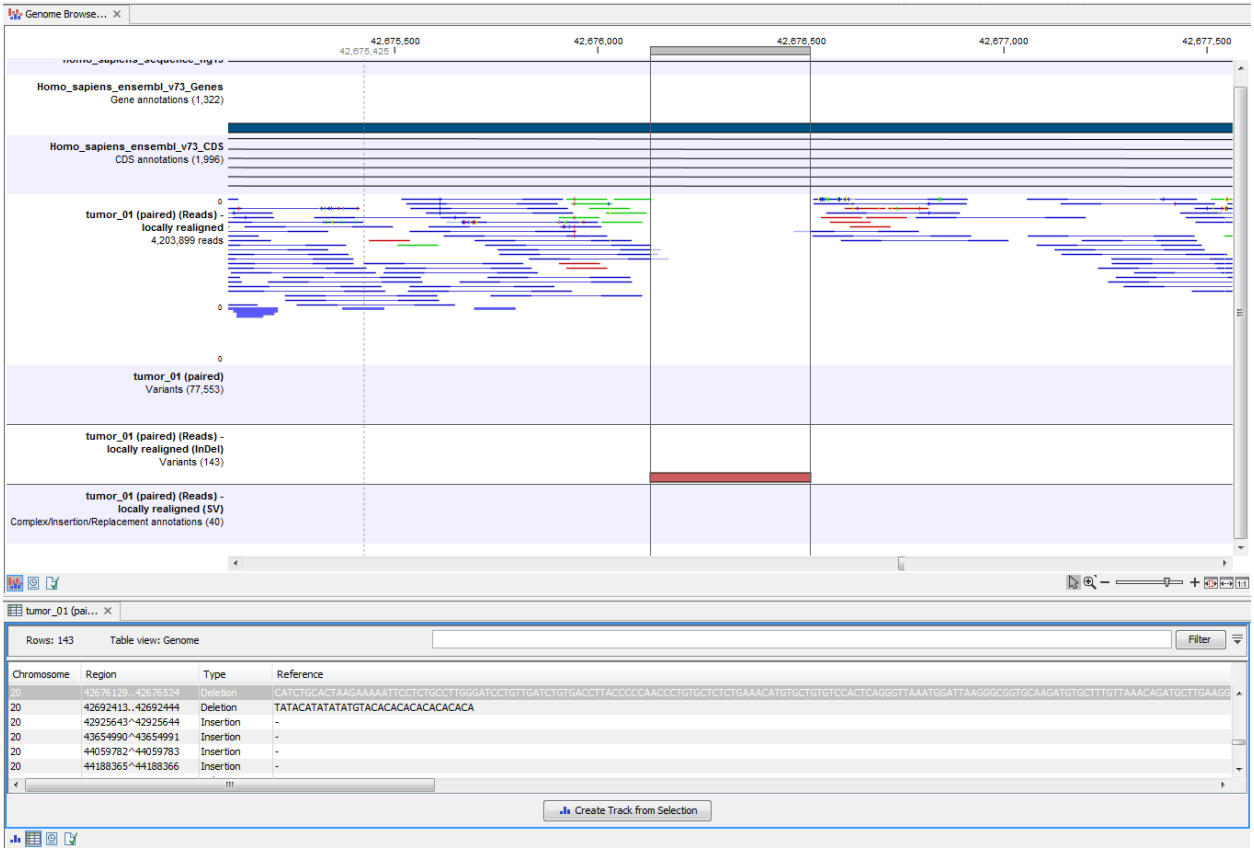


Figure 15.30: *This figure shows a Genome Browser View with an open track table. The table allows deeper inspection of the identified variants.*

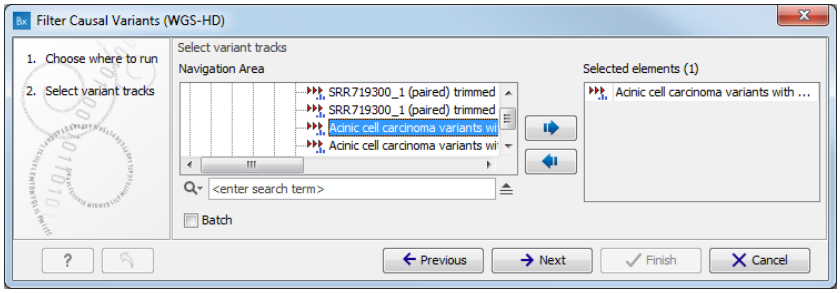


Figure 15.31: *Select the variant track from which you would like to filter somatic variants.*

This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

6. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

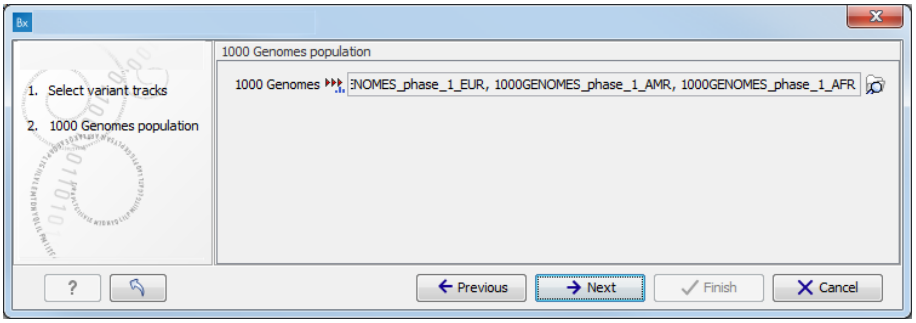


Figure 15.32: *Select the relevant 1000 Genomes population(s).*

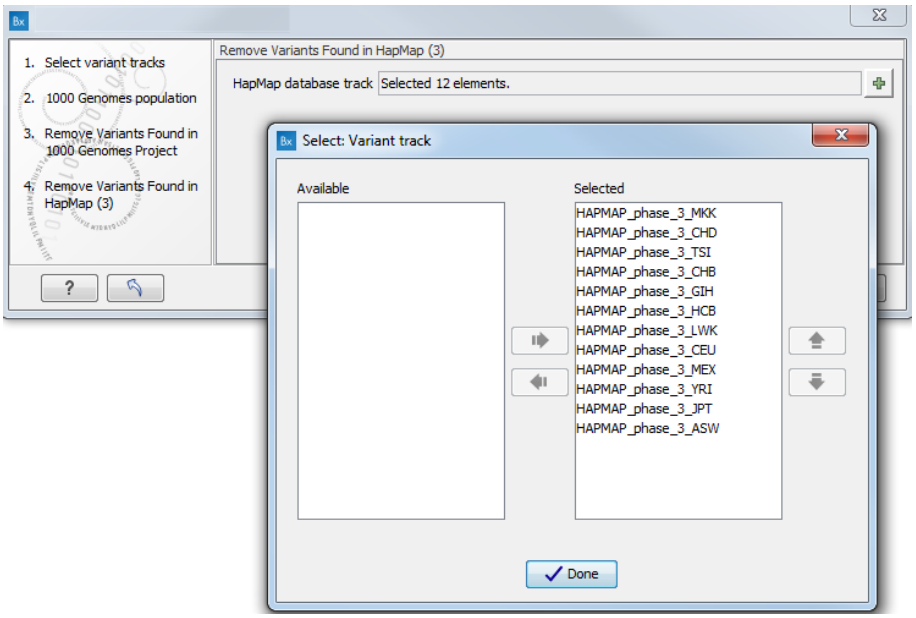


Figure 15.33: *Select the relevant Hapmap population(s).*

Output from the Filter Causal Variants (WGS-HD) workflow

Three types of output are generated:

- An **Amino Acid Track**
- A **Genome Browser View**
- A **Filtered Variant Track**

15.3.2 Identify Causal Inherited Variants in Family of Four (WGS)

As the name of the workflow implies, you can use the **Identify Causal Inherited Variants in a Family of Four (WGS)** ready-to-use workflow to identify inherited causal variants in a family of four. The family relationship can be a child, a mother, a father and one additional affected family member where, in addition to the child (the proband) one of the parents are affected and one additional family member is affected. The fourth family member can be any related and affected family member such as a sibling, grand parent, uncle or the like.

The **Identify Causal Inherited Variants in a Family of Four (WGS)** ready-to-use workflow accepts sequencing reads as input from each of the four family members.

How to run the Identify Causal Inherited Variants in a Family of Four (WGS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Family of Four (WGS)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Causal Inherited Variants in a Family of Four (WGS)** ( )

1. Double-click on the **Identify Causal Inherited Variants in a Family of Four (WGS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 15.34).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

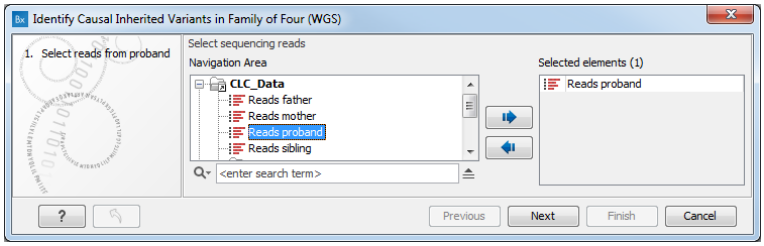


Figure 15.34: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **affected parent**.
4. Select the sequencing reads from the **unaffected parent**.
5. Select the sequencing reads from the **affected family member**.
6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 15.35).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

   - **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant

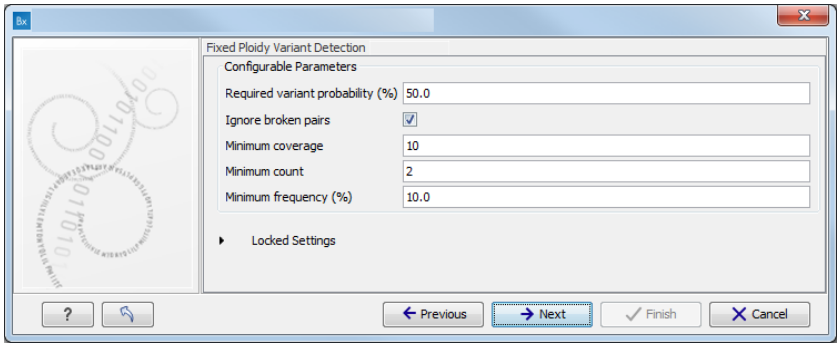


Figure 15.35: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected family member**.
10. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 15.36).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).
11. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

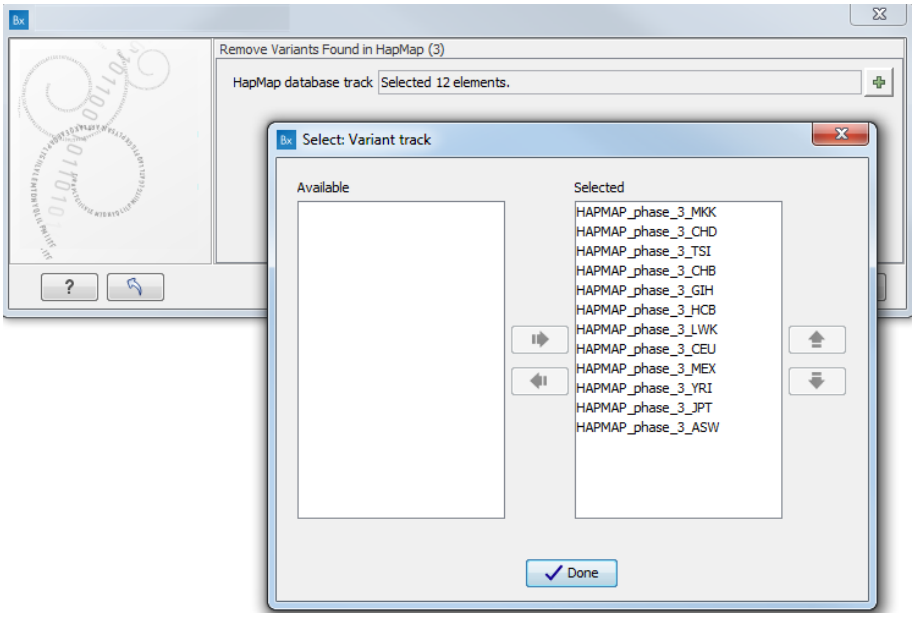


Figure 15.36: *Select the relevant Hapmap population(s).*

Output from the Identify Causal Inherited Variants in a Family of Four (WGS) workflow

Five types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified putative causal variants with the read mappings and information from databases.

15.3.3 Identify Causal Inherited Variants in Trio (WGS)

The **Identify Causal Inherited Variants in a Trio (WGS)** ready-to-use workflow identifies putative disease causing inherited variants by creating a list of variants present in both affected individuals and subtracting all variants in the unaffected individual. The workflow includes a back-check for all family members

The **Identify Causal Inherited Variants in a Trio (WGS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Causal Inherited Variants in a Trio (WGS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Trio (WGS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Causal Inherited Variants in a Trio (WGS)** ( )

1. Double-click on the **Identify Causal Inherited Variants in a Trio (WGS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 15.37).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

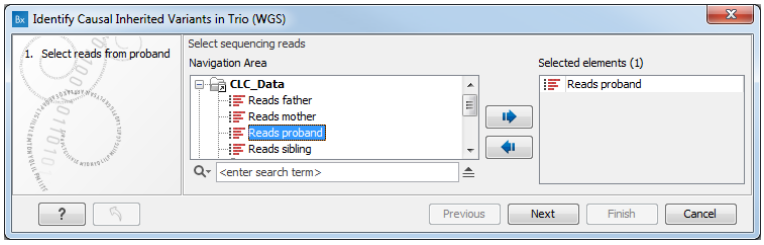


Figure 15.37: *Specify the sequencing reads for the appropriate family member.*

3. Select the reads for the **affected parent**.
4. Select the reads for the **unaffected parent**.
5. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 15.38).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

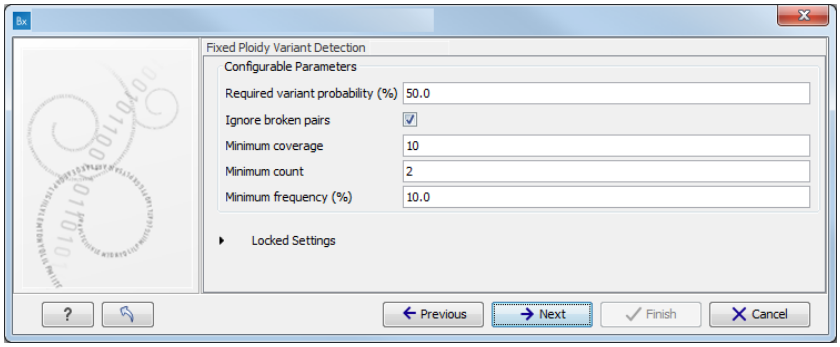


Figure 15.38: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
8. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 15.39).

   This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).
9. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

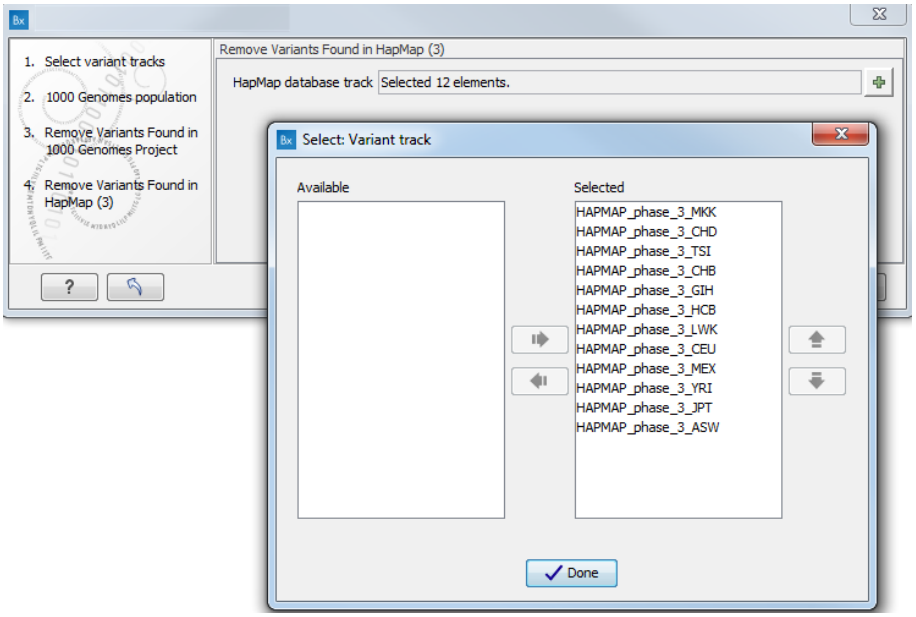


Figure 15.39: *Select the relevant Hapmap population(s).*

Output from the Identify Causal Inherited Variants in a Trio (WGS) workflow

Five types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

15.3.4 Identify Rare Disease Causing Mutations in Family of Four (WGS)

You can use the **Identify Rare Disease Causing Mutations in a Family of Four (WGS)** ready-to-use workflow to identifie de novo and compound heterozygous variants from an extended family of four, where the fourth individual is not affected.

The **Identify Rare Disease Causing Mutations in a Family of Four (WGS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Family of Four (WGS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Family of Four (WGS)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Genome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Rare Disease Causing Mutations in a Family of Four (WGS**
( )

1. Double-click on the **Identify Rare Disease Causing Mutations in a Family of Four (WGS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 15.40).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

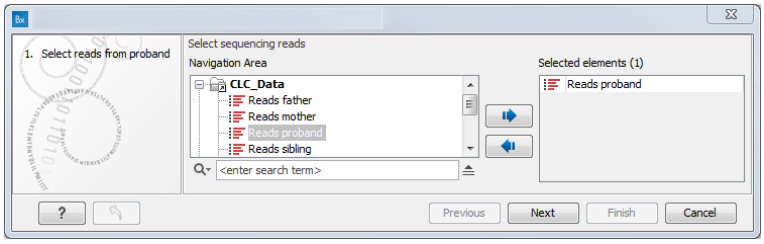


Figure 15.40: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **mother**.
4. Select the sequencing reads from the **father**.
5. Select the sequencing reads from the **unaffected sibling**.
6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 15.41).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

   - **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the

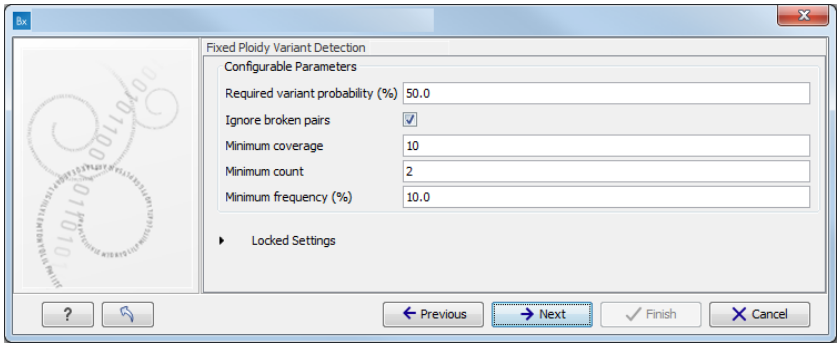


Figure 15.41: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

7. Specify the **Fixed Ploidy Variant Detection** settings that should be used for the for the **mother**.
8. Specify the **Fixed Ploidy Variant Detection** settings that should be used for the for the **father**.
9. Specify the **Fixed Ploidy Variant Detection** settings that should be used for the for the **unaffected sibling**.
10. Specify the affected child's **gender** (figure 15.42)
11. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **proband** (figure 15.43).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

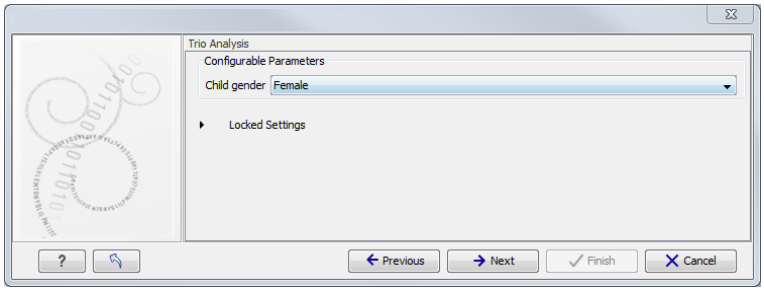


Figure 15.42: *Specify the proband's gender.*

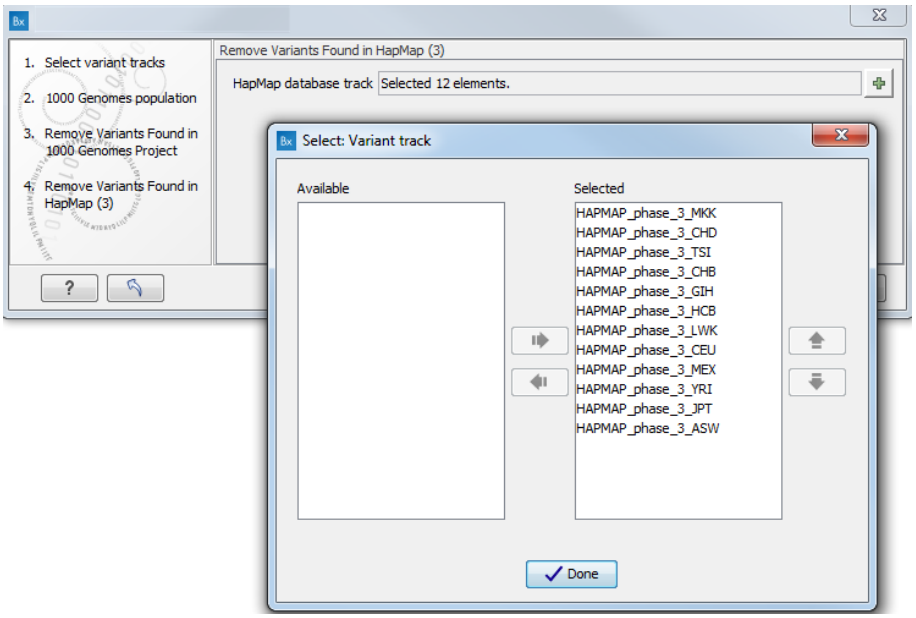


Figure 15.43: *Select the relevant Hapmap population(s).*

12. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father**.

13. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.

14. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.

15. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Rare Disease Causing Mutations in a Family of Four (WGS) workflow

Eleven types of output are generated:

- **Read Mapping** One for each family member. The reads mapped to the reference sequence.

- **Filtered Variant Track** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Read Mapping Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **De novo variants** Variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with de novo Variants** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**

15.3.5 Identify Rare Disease Causing Mutations in Trio (WGS)

The **Identify Rare Disease Causing Mutations in a Trio (WGS)** identifies de novo and compound heterozygous variants from a Trio. The workflow includes a back-check for all family members.

The **Identify Rare Disease Causing Mutations in a Trio (WGS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Trio (WGS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Trio (WGS)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Genome Sequencing** () | **Hereditary Disease** () | **Identify Rare Disease Causing Mutations in a Trio (WGS** ()

1. Double-click on the **Identify Rare Disease Causing Mutations in a Trio (WGS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 15.44).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

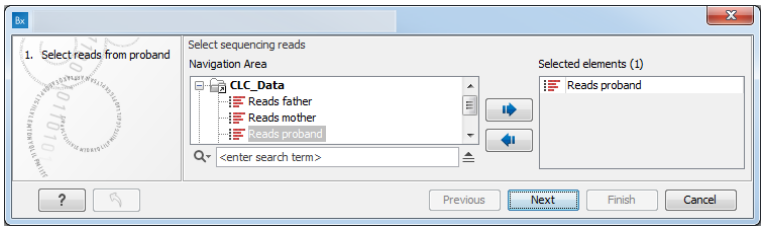


Figure 15.44: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **mother**.

4. Select the sequencing reads from the **father**.

5. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 15.45).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

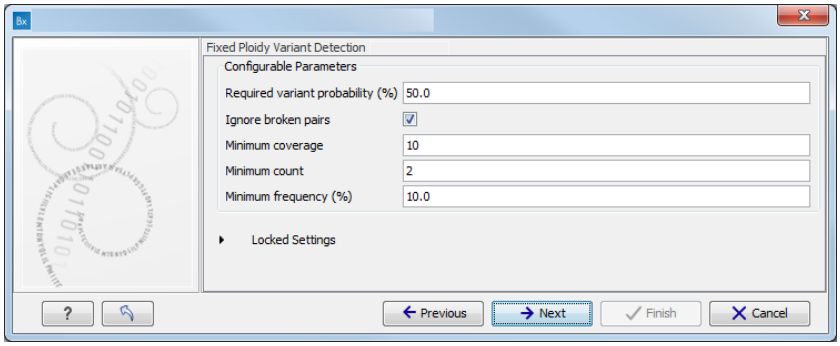


Figure 15.45: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required

variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

6. Set up the parameters for the **Fixed Ploidy Variant Detection** tool for the **mother**.
7. Set up the parameters for the **Fixed Ploidy Variant Detection** tool for the **father**.
8. Specify the affected child's **gender** (figure 15.46).

   Some workflows take the gender into account. When asked for it, provide the gender of the child (the proband).

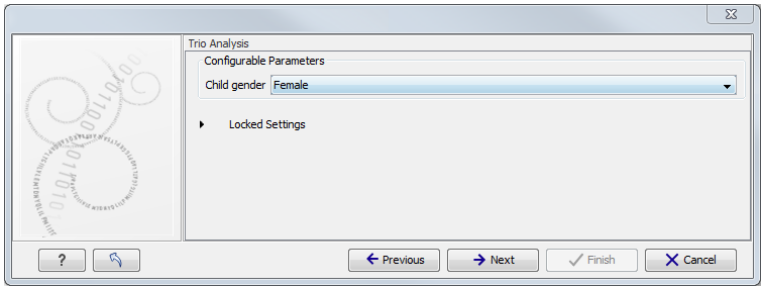


Figure 15.46: *Specify the proband's gender.*

9. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father** (figure 15.47).

   This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).
10. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **mother**.
11. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.
12. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.

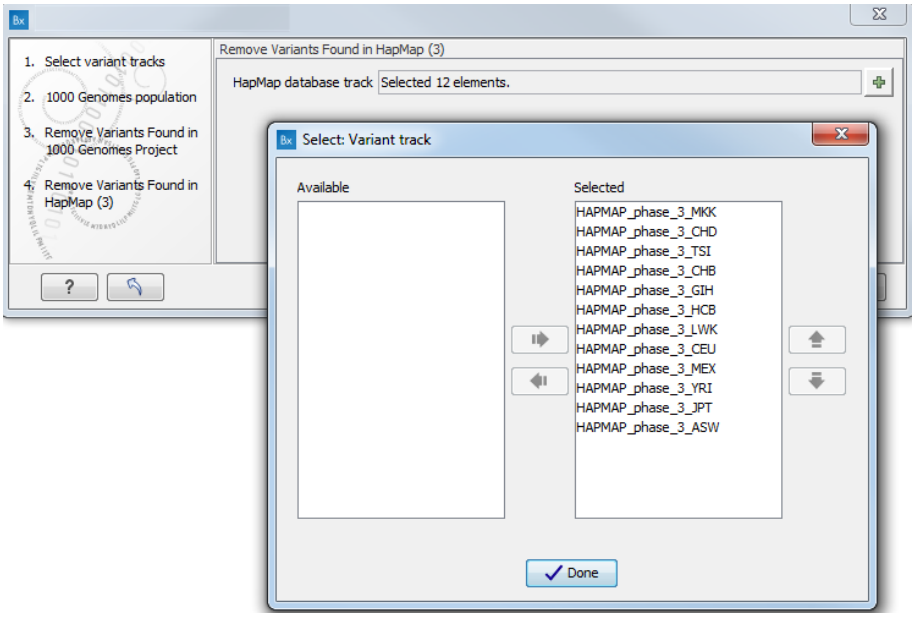


Figure 15.47: *Select the relevant Hapmap population(s).*

13. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Rare Disease Causing Mutations in a Trio (WGS) workflow

Eleven types of output are generated:

- **Read Mapping** One for each family member. The reads mapped to the reference sequence.
- **Filtered Variant Tracks** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Read Mapping Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **De novo variants** Filtered variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Filtered variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.

- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**

15.3.6 Identify Variants (WGS-HD)

You can use the **Identify Variants (WGS-HD)** ready-to-use workflow to call variants in the mapped and locally realigned reads. The workflow removes false positives and, in case of a targeted experiment, removes variants outside the targeted region. Variant calling is performed with the Fixed Ploidy Variant Detection tool.

The **Identify Variants (WGS-HD)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Variants (WGS-HD) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Variants (WGS-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Genome Sequencing** () | **Hereditary Disease** () | **Identify Variants (WGS-HD)** ()

1. Double-click on the **Identify Variants (WGS-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads you want to analyze (figure 15.48). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

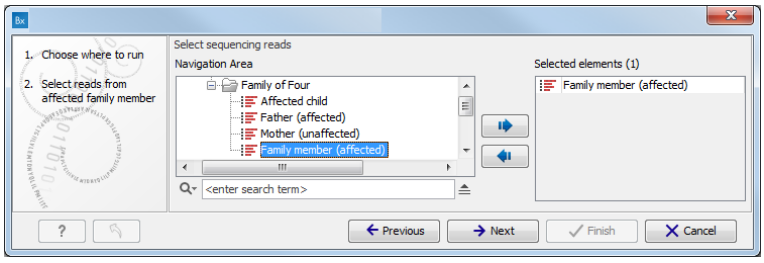


Figure 15.48: *Specify the sequencing reads for the appropriate family member.*

3. Specify the parameters for the **Fixed Ploidy Variant Detection** tool, including a target region file (figure 15.49).

The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

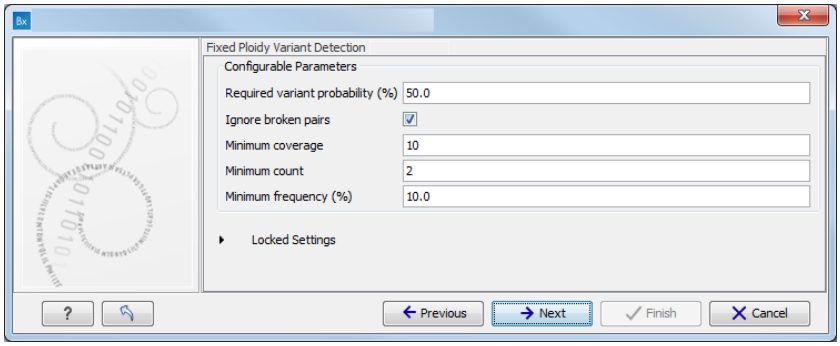


Figure 15.49: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

4. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Variants (WGS-HD) workflow

Six types of output are generated:

- A **Structural Variants**
- A **Structural Variants Report**
- A **Reads Track** Read Mapping
- A **Filtered Variant Track** Identified variants
- A **Read Mapping Report**
- A **Genome Browser View**

Chapter 16

Whole exome sequencing (WES)

Contents

The protein coding part of the human genome accounts for around 1 % of the genome and consists of around 180,000 exons covering an area of 30 megabases (Mb) [Ng et al., 2009]. By targeting sequencing to only the protein coding parts of the genome, exome sequencing is a cost efficient way of generating sequencing data that is believed to harbor the vast majority of the disease-causing mutations [Choi et al., 2009].

Thirteen ready-to-use workflows are available for analysis of whole genome sequencing data (figure 16.1). The concept of the pre-installed ready-to-use workflows is that read data are used as input in one end of the workflow and in the other end of the workflow you get a track based genome browser view and a table with all the identified variants, which may or may not have been subjected to different kinds of filtering and/or annotation.

In this chapter we will discuss what the individual ready-to-use workflows can be used for and go through step by step how to run the workflows.

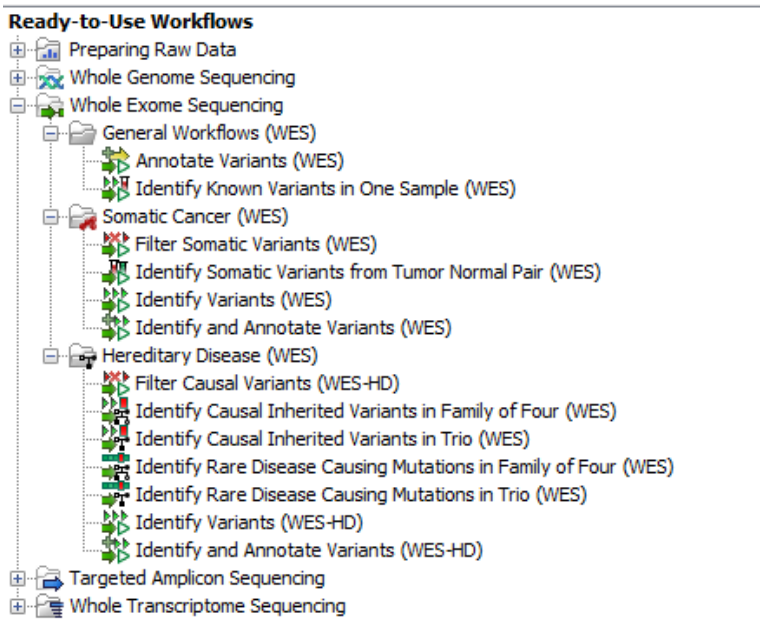


Figure 16.1: *The eleven workflows available for analyzing whole exome sequencing data.*

Note! Often you will have to prepare data with one of the two **Preparing Raw Data** workflows described in section 14 before you proceed to **Analysis of sequencing data (WES)**.

16.1 General Workflows (WES)

16.1.1 Annotate Variants (WES)

Using a variant track ( ) (e.g. the output from the Identify Variants ready-to-use workflow) the **Annotate Variants (WES)** ready-to-use workflow runs an internal workflow that adds the following annotations to the variant track:

- **Gene names** Adds names of genes whenever a variant is found within a known gene.
- **mRNA** Adds names of mRNA whenever a variant is found within a known transcript.
- **CDS** Adds names of CDS whenever a variant is found within a coding sequence.
- **Amino acid changes** Adds information about amino acid changes caused by the variants.
- **Information from ClinVar** Adds information about the relationships between human variations and their clinical significance.
- **Information from dbSNP** Adds information from the "Single Nucleotide Polymorphism Database", which is a general catalog of genome variation, including SNPs, multinucleotide polymorphisms (MNPs), insertions and deletions (indels), and short tandem repeats (STRs).
- **PhastCons Conservation scores** The conservation scores, in this case generated from a multiple alignment with a number of vertebrates, describe the level of nucleotide conservation in the region around each variant.

How to run the Annotate Variants (WES) workflow

1. Go to the toolbox and select the **Annotate Variants (WES)** workflow. In the first wizard step, select the input variant track (figure 16.2).

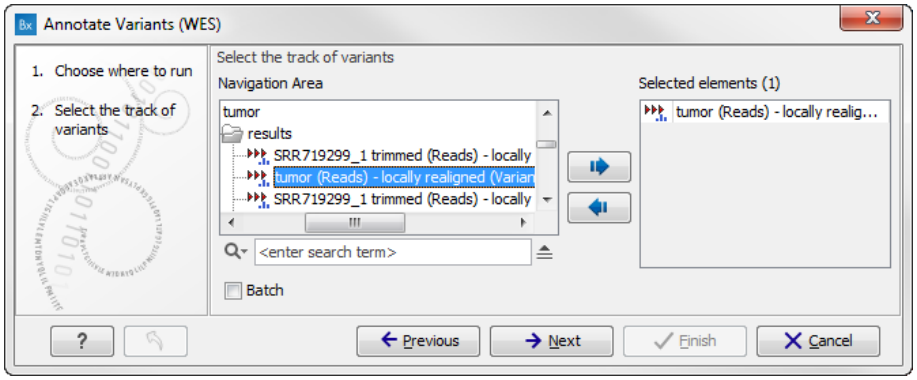


Figure 16.2: *Select the variant track to annotate.*

2. Click on the button labeled **Next**. The only parameter that should be specified by the user is which 1000 Genomes population yo use (figure 16.3). This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

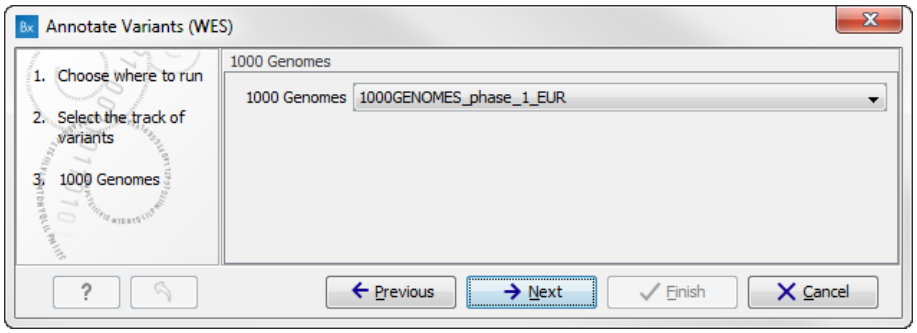


Figure 16.3: *Select the relevant 1000 Genomes population(s).*

3. Click on the button labeled **Next** to go to the last wizard step (figure 16.4).

   In this wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

4. Choose to **Save** your results and click on the button labeled **Finish**.

Output from the Annotate Variants (WES) workflow

Two types of output are generated:

1. **Annotated Variants** ( ) Annotation track showing the variants. Hold the mouse over one

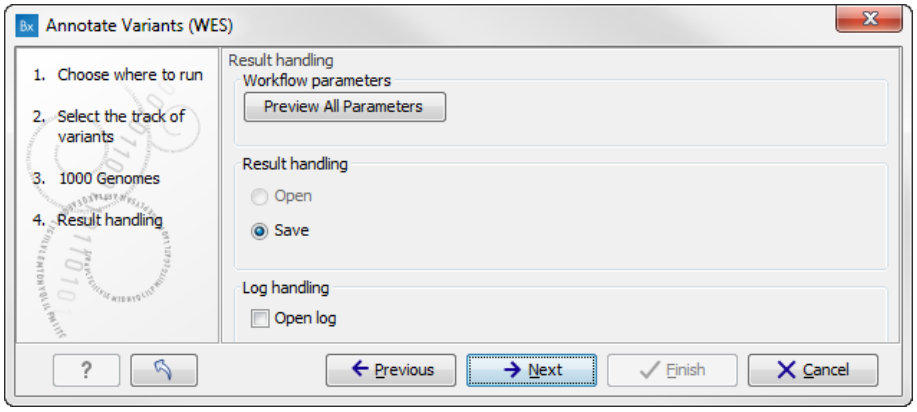


Figure 16.4: *Check the settings and save your results.*

of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant.

2. An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.

3. **Genome Browser View Annotated Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in dbSNP, ClinVar, 1000 Genomes, and PhastCons conservation scores (see figure 16.5).

Note! Please be aware, that if you delete the annotated variant track, this track will also disappear from the genome browser view.

It is possible to add tracks to the **Genome Browser View** such as mapped sequencing reads as well as other tracks. This can be done by dragging the track directly from the **Navigation Area** to the **Genome Browser View**.

If you double-click on the name of the annotated variant track in the left hand side of the Genome Browser View, a table that includes all variants and the added information/annotations will open (see figure 16.6). The table and the Genome Browser View are linked; if you click on an entry in the table, this particular position in the genome will automatically be brought into focus in the Genome Browser View.

You may be met with a warning as shown in figure 16.7. This is simply a warning telling you that it may take some time to create the table if you are working with tracks containing large amounts of annotations. Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar database, can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

Figure 16.5: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list) containing individual tracks for all added annotations.*

A high conservation level between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** ( ) | **Create Filter Criteria** ( )

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used

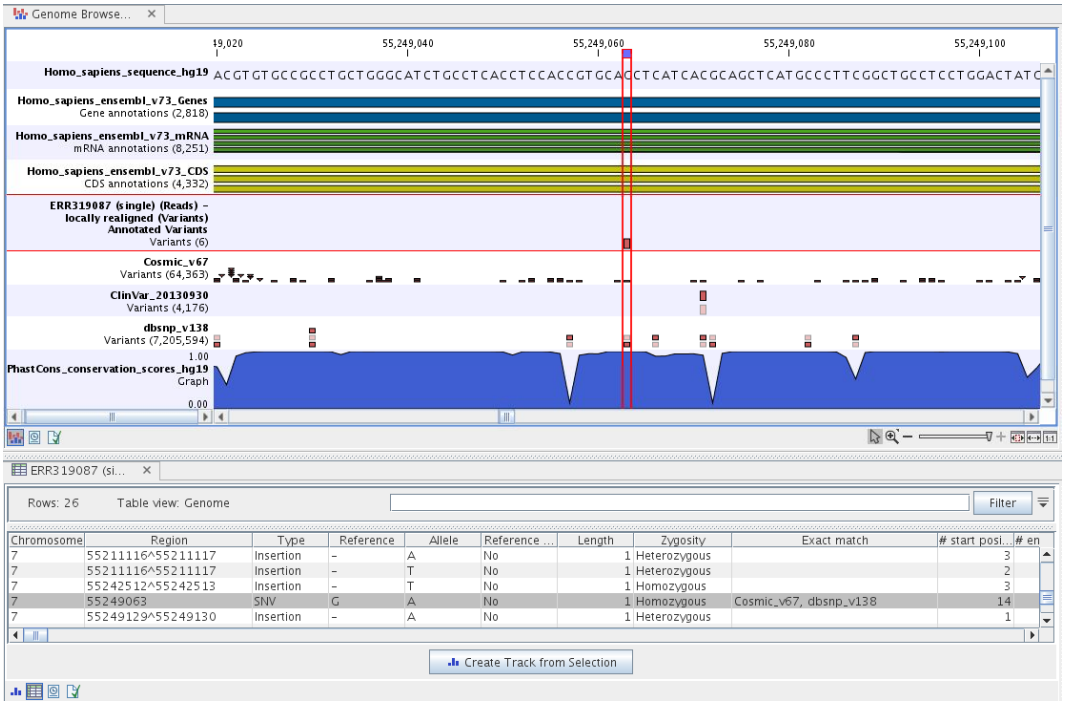


Figure 16.6: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list). The information is also available in table view. Click on the small table icon to open the table view. If you hold down the "Ctrl" key while clicking on the table icon, you will open a split view showing both the genome browser view and the table view.*

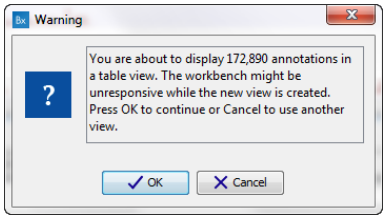


Figure 16.7: *Warning that appears when you work with tracks containing many annotations.*

databases. This can be done with "Data Management" function, which is described in section 13.1.

16.1.2 Identify Known Variants in One Sample (WES)

The **Identify Known Variants in One Sample (WES)** ready-to-use workflow is a combined data analysis and interpretation ready-to-use workflow.

It should be used to identify known variants specified by the user (e.g. known breast cancer associated variants) for their presence or absence in a sample.

Please note that the ready-to-use workflow will not identify new variants.

The **Identify Known Variants in One Sample (WES)** ready-to-use workflow maps the sequencing reads to a human genome sequence and does a local realignment of the mapped reads to improve the subsequent variant detection. In the next step, only variants specified by the user

are identified and annotated in the newly generated read mapping.

Import your known variants

To make an import into the *Biomedical Genomics Workbench*, you should have your variants in GVF format (http://www.sequenceontology.org/resources/gvf.html or VCF format http://ga4gh.org/#/fileformats-team).

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit will be provided by the vendor. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get it in either .bed or .gff format.

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

How to run the Identify Known Variants in One Sample (WES) workflow

1. Go to the toolbox and double-click on

   Toolbox | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **General Workflows (WES)** | **Identify Known Variants from One Sample (WES)** ( )

2. This will open the wizard step shown in figure 16.8 where you can select the reads of the sample that should be tested for presence or absence of your known variants.

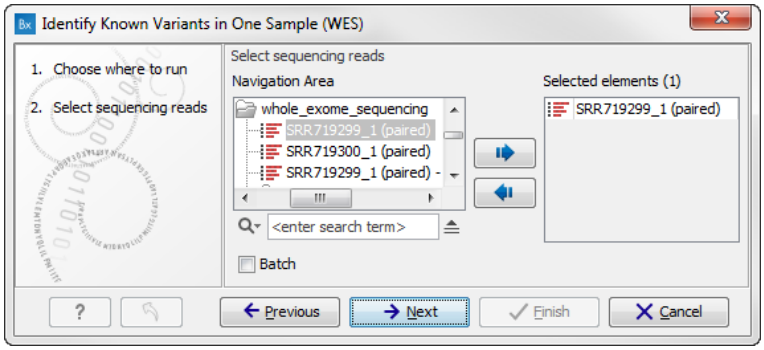


Figure 16.8: *Select the sequencing reads from the sample you would like to test for your known variants.*

   If several samples from different folders should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" and specifying the folders that hold the data you wish to analyse.

   Click on the button labeled **Next**.

3. Specify the target region for the Indels and Structural Variants tool (figure 16.9). This step is optional and will speed the completion time of the workflow by running the tool only on

the selected target regions. If you do not have a targeted region file to provide, simply click **Next**.

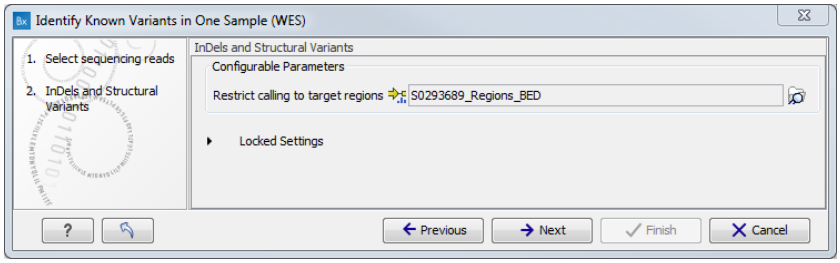


Figure 16.9: *Specify the targeted region file for the Indels and Structural Variants tool.*

4. Specify the parameters for the QC for Target Sequencing tool (figure 16.10).

   When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. This step is not optional, and you need to specify the targeted regions file adapted to the sequencing technology you used. Choose to use the default settings or to adjust the parameters.

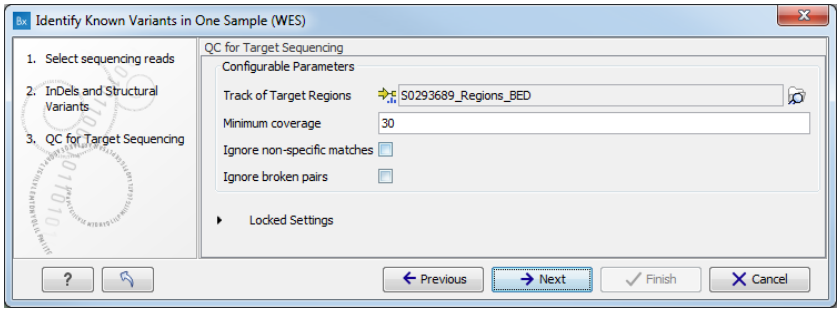


Figure 16.10: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

For more information about the tool, see section 20.1.

5. Click on the button labeled **Next** and specify the track with the known variants that should be identified in your sample (figure 16.11).

   The parameters that can be set are:

   - **Minimum coverage** The minimum number of reads that covers the position of the variant, which is required to set "Sufficient Coverage" to YES.
   - **Detection frequency** The minimum allele frequency that is required to annotate a variant as being present in the sample. The same threshold will also be used to determine if a variant is homozygous or heterozygous. In case the most frequent alternative allele at the position of the considered variant has a frequency of less than this value, the zygosity of the considered variant will be reported as being homozygous.

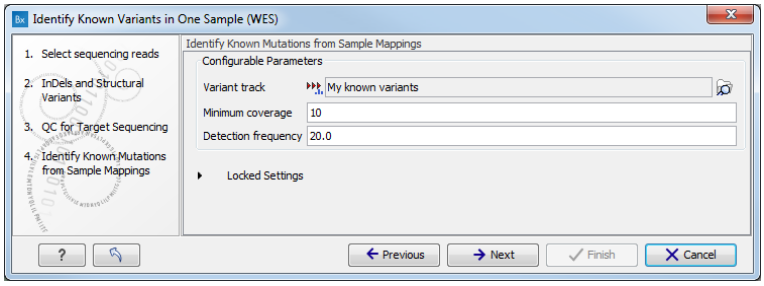


Figure 16.11: *Specify the track with the known variants that should be identified.*

The parameter "Detection Frequency" will be used in the calculation twice. First, it will report in the result if a variant has been detected (observed frequency > specified frequency) or not (observed frequency <= specified frequency). Moreover, it will determine if a variant should be labeled as heterozygous (frequency of another allele identified at a position of a variant in the alignment > specified frequency) or homozygous (frequency of all other alleles identified at a position of a variant in the alignment < specified frequency).

Click on the button labeled **Next**.

6. In the last wizard step (figure 16.12)you can check the selected settings by clicking on the button labeled **Preview All Parameters**.

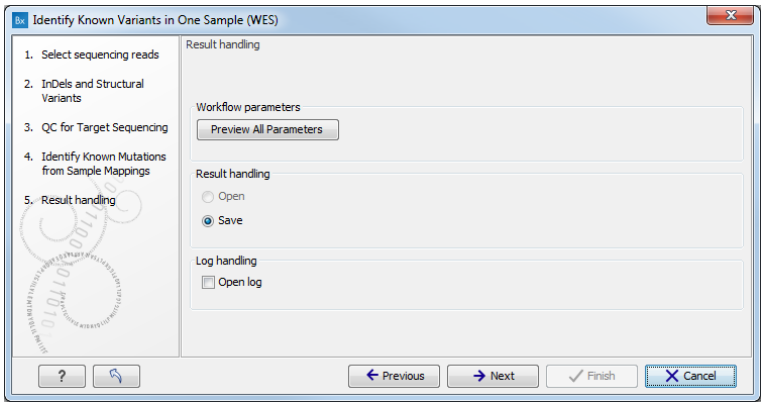


Figure 16.12: *Check the settings and save your results.*

At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination.

7. Click on the button labeled **OK** to go back to the previous dialog box and choose to **Save** your results.

   Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Known Variants in One Sample (WES)

The **Identify Known Variants in One Sample (WES)** tool produces five different output types:

- **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).
- **Target Regions Coverage** ( ) A track showing the targeted regions. The table view provides information about the targeted regions such as target region length, coverage, regions without coverage, and GC content.
- **Target Regions Coverage Report** ( ) The report consists of a number of tables and graphs that in different ways show e.g. the number, length, and coverage of the target regions and provides information about the read count per GC%.
- **Variants Detected in Detail** ( ) Annotation track showing the known variants. Like the "Overview Variants Detected" table, this table provides information about the known variants. Four columns starting with the sample name and followed by "Read Mapping coverage", "Read Mapping detection", "Read Mapping frequency", and "Read Mapping zygosity" provides the overview of whether or not the known variants have been detected in the sequencing reads, as well as detailed information about the Most Frequent Alternative Allele (labeled MFAA).
- **Genome Browser View Identify Known Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, target regions coverage, the mapped reads, the overview of the detected variants, and the variants detected in detail.

It is a good idea to start looking at the Target Regions Coverage Report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Please also check that at least 90% of the reads are mapped to the human reference sequence. In case of a targeted experiment, we also recommend that you check that the majority of the reads are mapping to the targeted region.

When you have inspected the target regions coverage report you can open the Genome Browser View Identify Known Variants file (see 16.13).

The Genome Browser View includes an overview track of the known variants and a detailed result track presented in the context of the human reference sequence, genes, transcripts, coding regions, targeted regions, and mapped sequencing reads.

Finally, a track with conservation scores has been added to be able to see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant.

By double clicking on one of the annotated variant tracks in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see 16.14).

Note We do not recommend that any of the produced files are deleted individually as some of them are linked to other outputs. Please always delete all of them at the same time.

16.2 Somatic Cancer (WES)

16.2.1 Filter Somatic Variants (WES)

If you are analyzing a list of variants that have been detected in a tumor or blood sample where no control sample is available from the same patient, you can use the **Filter Somatic**

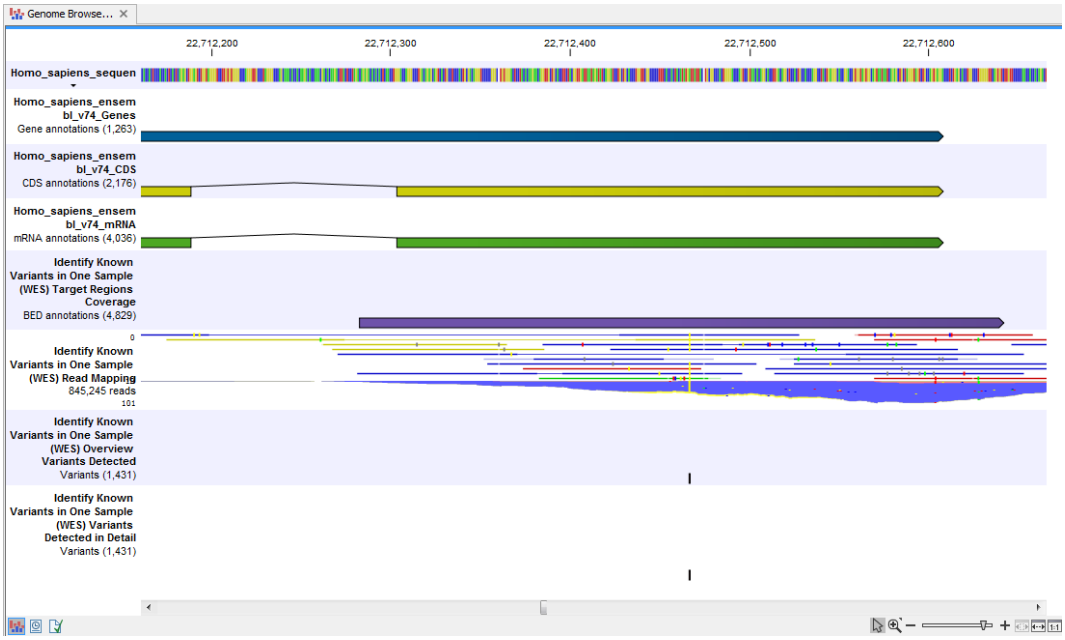


Figure 16.13: *Genome Browser View that allows inspection of the identified variants in the context of the human genome and external databases.*

Variants (WES) ready-to-use workflow to identify potential somatic variants. The purpose of this ready-to-use workflow is to use publicly available (or your own) databases, with common variants in a population, to extract potential somatic variants whenever no control/normal sample from the same patient is available.

The **Filter Somatic Variants (WES)** ready-to-use workflow accepts variant tracks ( ) (e.g. the output from the Identify Variants ready-to-use workflow) as input. In cases with heterozygous variants, the reference allele is first filtered away, then variants outside the targeted region are removed, and lastly, variants found in the Common dbSNP, 1000 Genomes Project, and HapMap databases are deleted. Variants in those databases are assumed to not contain relevant somatic variants.

Please note that this tool will likely also remove inherited cancer variants that are present at a low percentage in a population.

Next, the remaining somatic variants are annotated with gene names, amino acid changes, conservation scores and information from ClinVar (known variants with medical impact) and dbSNP (all known variants).

How to run the Filter Somatic Variants (WES) workflow

To run the **Filter Somatic Variants (WES)** tool, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Somatic Cancer** ( ) | **Filter Somatic Variants** ( )

1. Double-click on the **Filter Somatic Variants** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

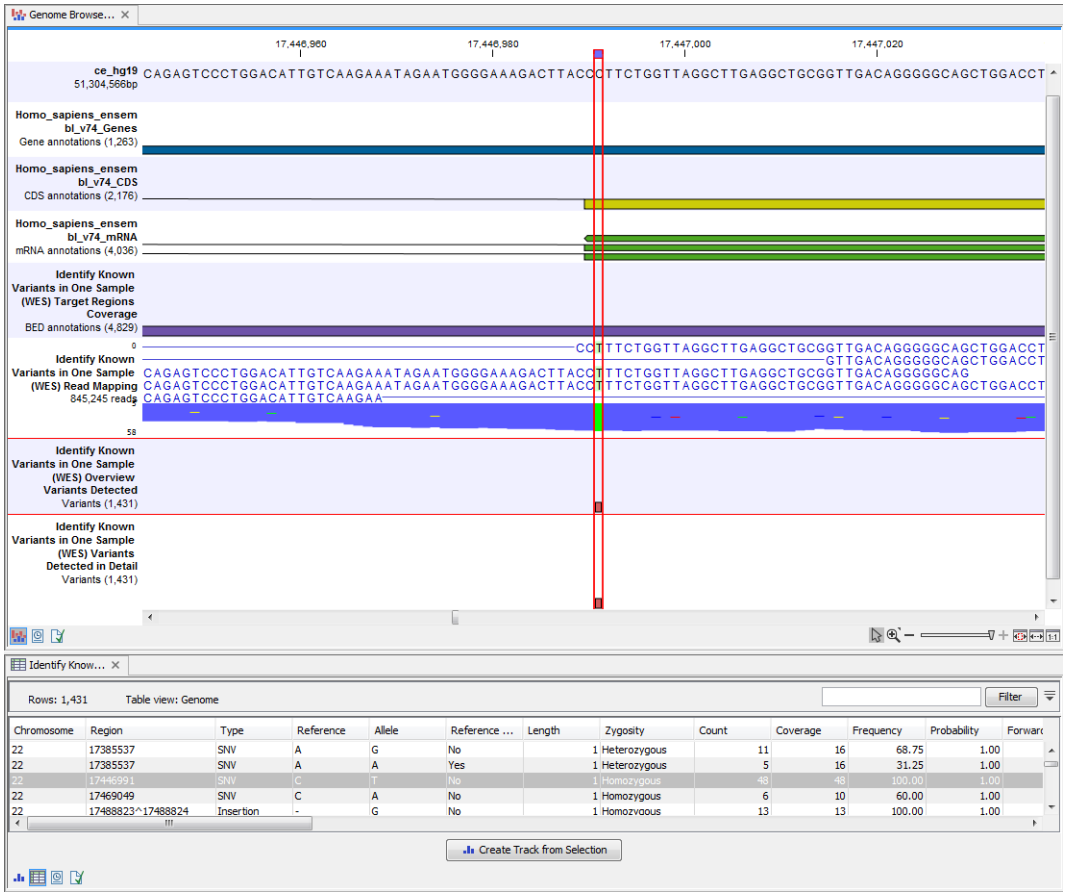


Figure 16.14: *Genome Browser View with an open overview variant track with information about if the variant has been detected or not, the identified zygosity, if the coverage was sufficient at this position and the observed allele frequency.*

2. Next, you will be asked to select the variant track you would like to use for filtering somatic variants. The panel in the left side of the wizard shows the kind of input that should be provided (figure 16.15). Select by double-clicking on the reads file name or clicking once on the file and then clicking on the arrow pointing to the right side in the middle of the wizard.

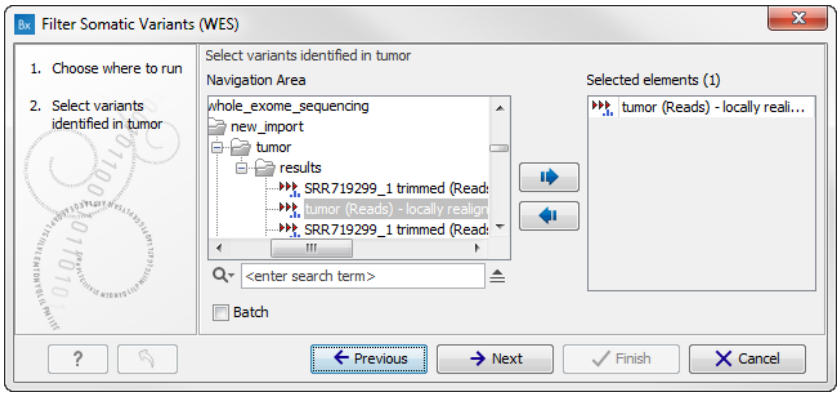


Figure 16.15: *Select the variant track from which you would like to filter somatic variants.*

Click on the button labeled **Next**.

3. In the next step you will be asked to specify which of the 1000 Genomes populations that should be used for annotation (figure 16.16).

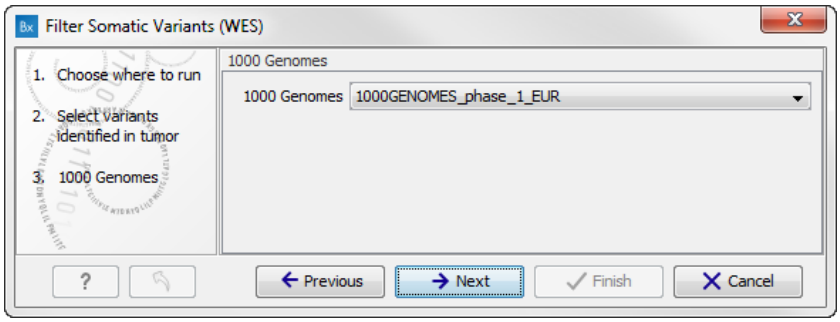


Figure 16.16: *Specify which 1000 Genomes population to use for annotation.*

Click on the button labeled **Next**.

4. The next wizard step will once again allow you to specify the 1000 Genomes population that should be used, this time for filtering out variants found in the 1000 Genomes project (figure 16.17).

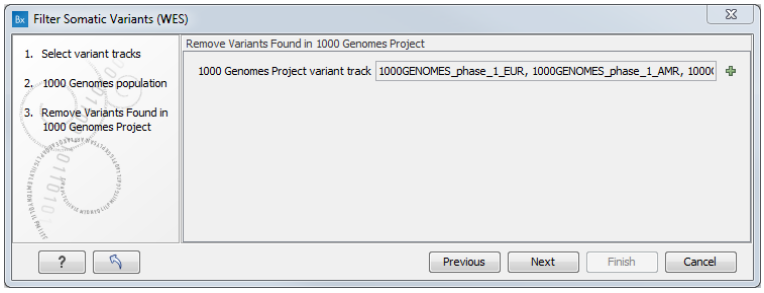


Figure 16.17: *Specify which 1000 Genomes population to use for filtering out known variants.*

Click on the button labeled **Next**.

5. The next wizard step (figure 16.18) concerns removal of variants found in the HapMap database. Select the population you would like to use from the drop-down list. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

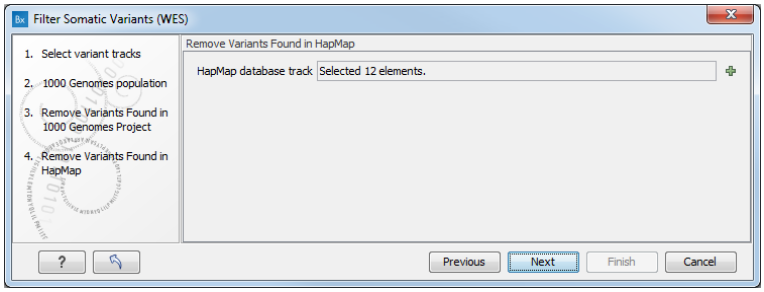


Figure 16.18: *Specify which HapMap population to use for filtering out known variants.*

6. Click on the button labeled **Next** to go to the last wizard step (shown in figure 16.19).

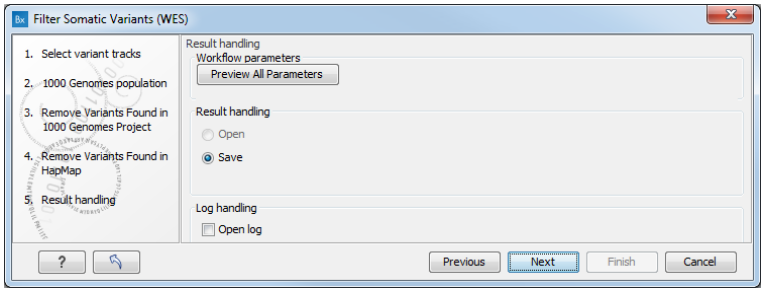


Figure 16.19: *Check the selected parametes by pressing "Preview All Parameters".*

Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to **Save** the results and click on the button labeled **Finish**.

Output from the Filter Somatic Variants (WES) workflow

Two types of output are generated:

- **Amino Acids Changes** Track that shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Somatic Candidate Variants** Track that holds the variant data. This track is also included in the Genome Browser View. If you hold down the Ctrl key (Cmd on Mac) while clicking on the table icon in the lower left side of the **View Area**, you can open the table view in split view. The table and the variant track are linked together, and when you click on a row in the table, the track view will automatically bring this position into focus.
- **Genome Browser View Filter Somatic Variants** A collection of tracks presented together. Shows the somatic candidate variants together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar, 1000 Genomes, and the PhastCons conservation scores (see figure 16.20).

To see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant, a track with conservation scores is added as well. Mapped sequencing reads as well as other tracks can be easily added to this Genome Browser View. By double clicking on the annotated variant track in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see figure 16.21).

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar database, can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level, between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting

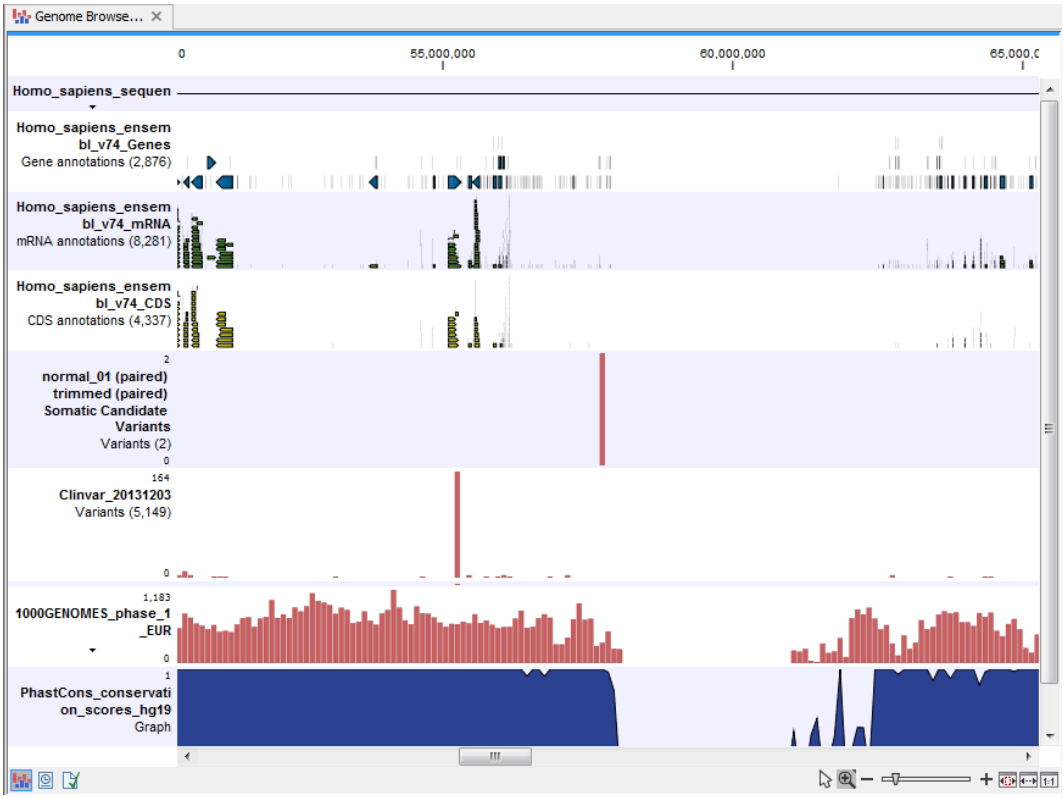


Figure 16.20: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** ( ) | **Create Filter Criteria** ( )

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

16.2.2 Identify Somatic Variants from Tumor Normal Pair (WES)

The **Identify Somatic Variants from Tumor Normal Pair (WES)** ready-to-use workflow can be used to identify potential somatic variants in a tumor sample when you also have a normal/control sample from the same patient.

When running the **Identify Somatic Variants from Tumor Normal Pair (WES)** the reads are mapped

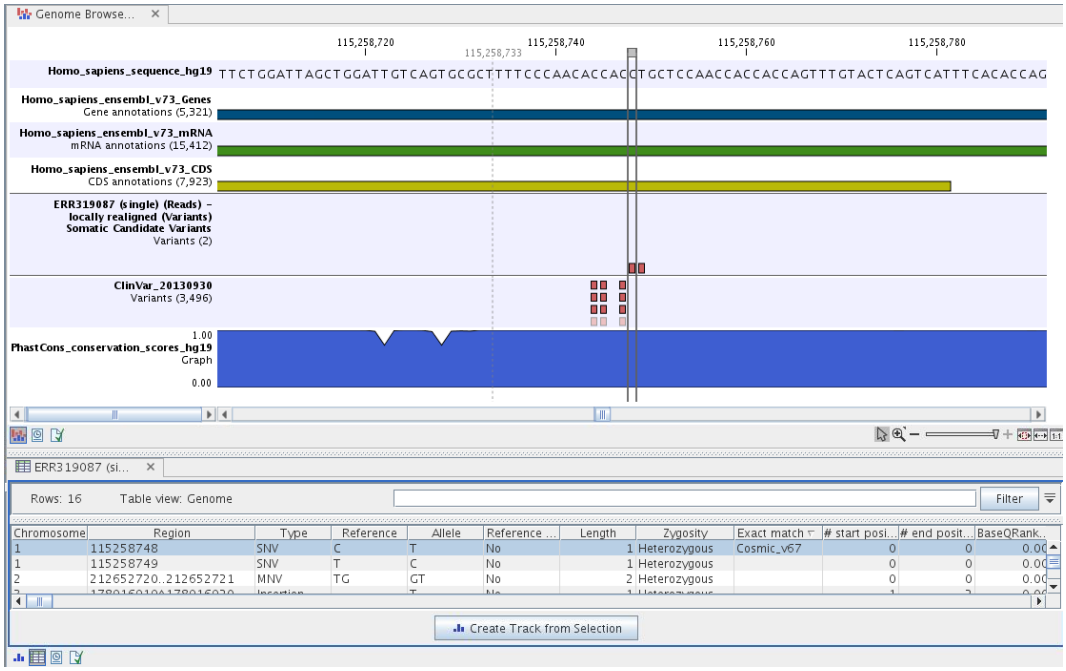


Figure 16.21: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

and the variants identified. An internal workflow removes germline variants that are found in the mapped reads of the normal/control sample and variants outside the target region are removed as they are likely to be false positives due to non-specific mapping of sequencing reads. Next, remaining variants are annotated with gene names, amino acid changes, conservation scores and information from relevant databases like ClinVar (variants with clinically relevant association). Finally, information from dbSNP is added to see which of the detected variants have been observed before and which are completely new.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit is available from the vendor of the enrichment kit and sequencing machine. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get the file in either .bed or .gff format.

To import the file:

Go to the toolbar | **Import (**  **)** | **Tracks (**  **)**

How to run the Identify Somatic Variants from Tumor Normal Pair (WES)

1. Go to the toolbox and double-click on the **Identify Somatic Variants from Tumor Normal Pair (WES)** ready-to-use workflow. This will open the wizard shown in figure 16.22 where you can select the tumor sample reads.

   When you have selected the tumor sample reads click on the button labeled **Next**.

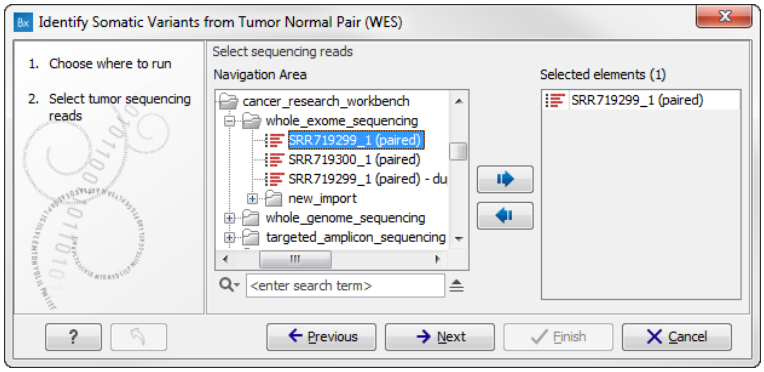


Figure 16.22: *Select the tumor sample reads.*

2. In the next wizard step (figure 16.23), please specify the normal sample reads.

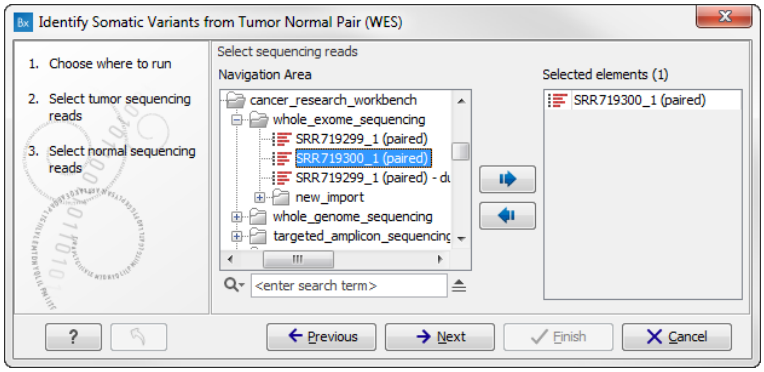


Figure 16.23: *Select the normal sample reads.*

3. The following 2 steps allow you to restrict the calling of indels and structural variants to the targeted regions, both for tumor and normal reads (figure 16.24).

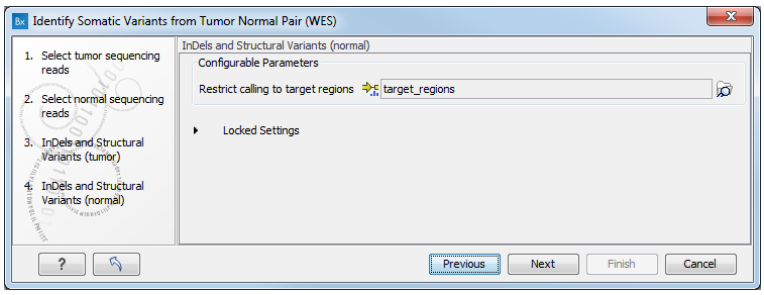


Figure 16.24: *Specify the target regions track.*

4. Set the parameters for the Low Frequency Variant Detection step (figure 16.25) and click **Next**.

5. In the following 2 wizard steps, you can select your target regions track to be used for reporting the performance of the targeted re-sequencing experiment for the tumor and normal samples successively (figure 16.26). The targeted region track should be the same as the track you selected in the previous wizard steps. Variants found outside the targeted regions will not be included in the output that is generated with the ready-to-use workflow.

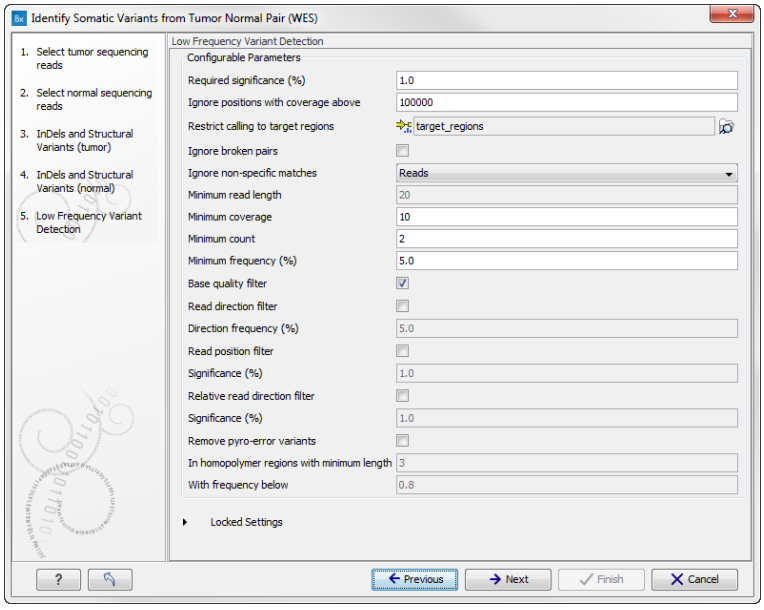


Figure 16.25: *Specify the settings for the variant detection.*

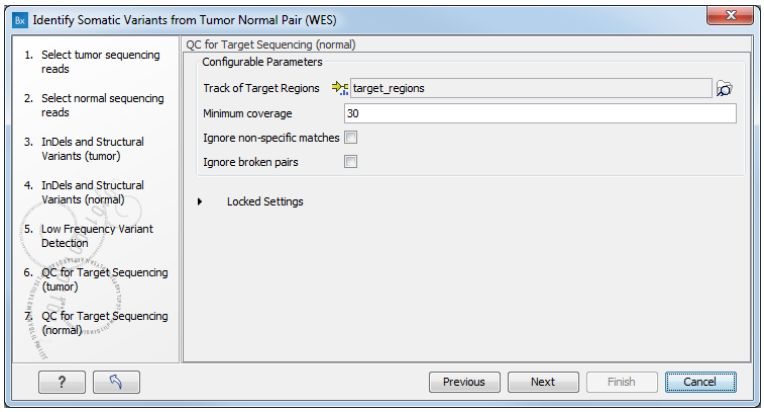


Figure 16.26: *Select your target region track.*

6. Next, adjust the settings for removal of germline variants step (figure 16.27). Click on the button labeled **Next**.

7. In the next wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters** (figure 16.28).

   In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

8. Click on the button labeled **OK** to go back to the previous wizard step and choose **Save**.

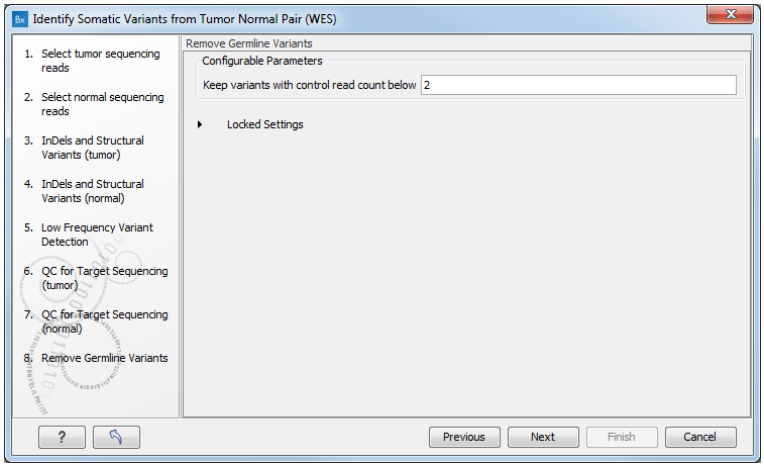


Figure 16.27: *Select your target region track.*

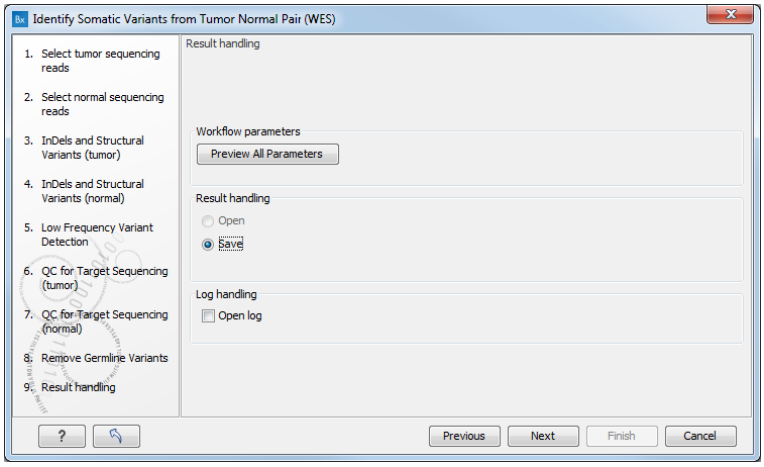


Figure 16.28: *Check the parameters and save the results.*

Output from the Identify Somatic Variants from Tumor Normal Pair (WES) workflow

Eight different outputs are generated:

1. **Read Mapping Normal** ( ) The mapped sequencing reads for the normal sample. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

2. **Read Mapping Tumor** ( ) The mapped sequencing reads for the tumor sample.

3. **Target Region Coverage Report Normal** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from the normal sample.

4. **Target Region Coverage Tumor** ( ) A track showing the targeted regions. The table view provides information about the targeted regions such as target region length, coverage, regions without coverage, and GC content.

5. **Target Region Coverage Report Tumor** ( ) The report consists of a number of tables and

graphs that in different ways provide information about the mapped reads from the tumor sample.

6. **Amino Acids Changes** Track that shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.

7. **Annotated Somatic Variants** ( ) A variant track holding the identified and annotated somatic variants. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

8. **Genome Browser View Tumor Normal Comparison** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads for both normal and tumor, the annotated somatic variants, information from the ClinVar database, and finally a track showing the conservation score (see figure 16.29).

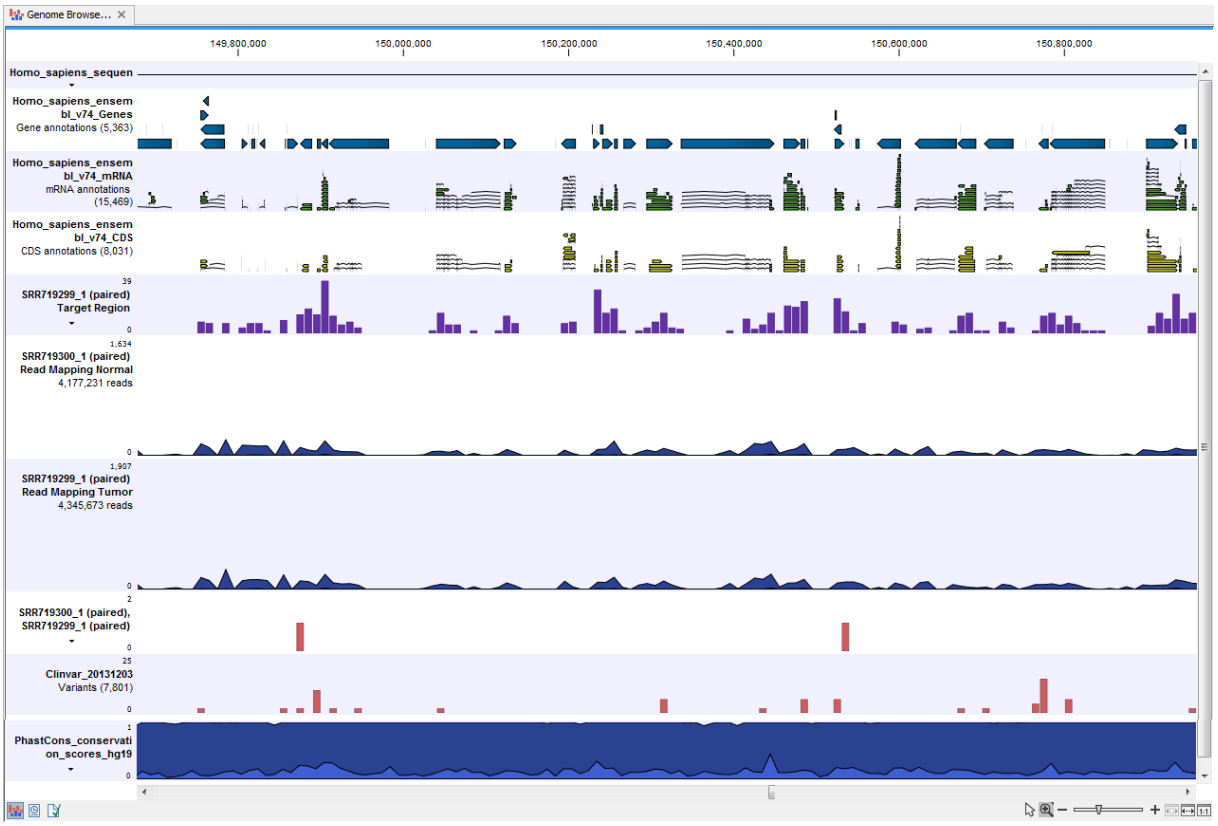


Figure 16.29: *The Genome Browser View presents all the different data tracks together and makes it easy to compare different tracks.*

16.2.3 Identify Variants (WES)

The **Identify Variants (WES)** tool takes sequencing reads as input and returns identified variants as part of a Genome Browser View.

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection, which is run afterwards. At the end, variants with an average base quality smaller than 20 are filtered away.

In addition, a targeted region report is created to inspect the overall coverage and mapping specificity in the targeted regions.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit will be provided by the vendor. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get it in either .bed or .gff format.

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

How to run the Identify Variants (WES) workflow

To run the **Identify Variants (WES)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Somatic Cancer** ( ) | **Identify Variants (WES)** ( )

1. Select the sequencing reads from the sample that should be analyzed (figure 16.30).

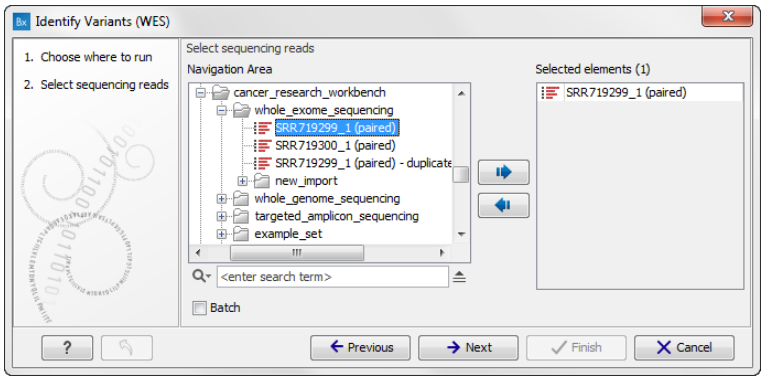


Figure 16.30: *Please select all sequencing reads from the sample to be analyzed.*

Select all sequencing reads from your sample. If several samples should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" (tick "Batch" at the bottom of the wizard as shown in figure 16.38) and select the **folder** that holds the data you wish to analyze. If you have your sequencing data in separate folders, you should choose to run the analysis in batch mode.

When you have selected the sample(s) you wish to prepare, click on the button labeled **Next**.

2. In this wizard you can restrict calling of indels and structural variants to the targeted regions by specifying the track with the targeted regions from the experiment (figure 16.31).

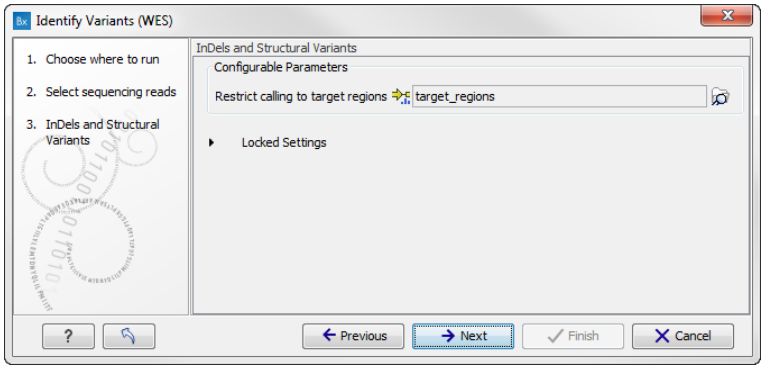


Figure 16.31: *Select the track with the targeted regions from your experiment.*

3. In the next wizard step (figure 16.32) you have to specify the track with the targeted regions from the experiment. You can also specify the minimum read coverage, which should be present in the targeted regions.

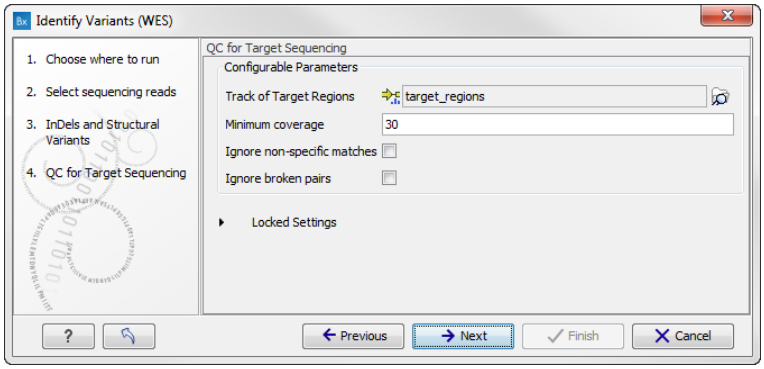


Figure 16.32: *Select the track with the targeted regions from your experiment.*

4. Click on the button labeled **Next**, which will take you to the next wizard step (figure 16.33). In this wizard you can specify the parameter for detecting variants.

5. Click on the button labeled **Next**, which will take you to the next wizard step (figure 16.34).

6. Click on the button labeled **Next** to go to the last wizard step (figure 16.35).

   In this wizard you get the chance to check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard step you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

7. Click on the button labeled **OK** to go back to the previous wizard step and choose **Save**.

   Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

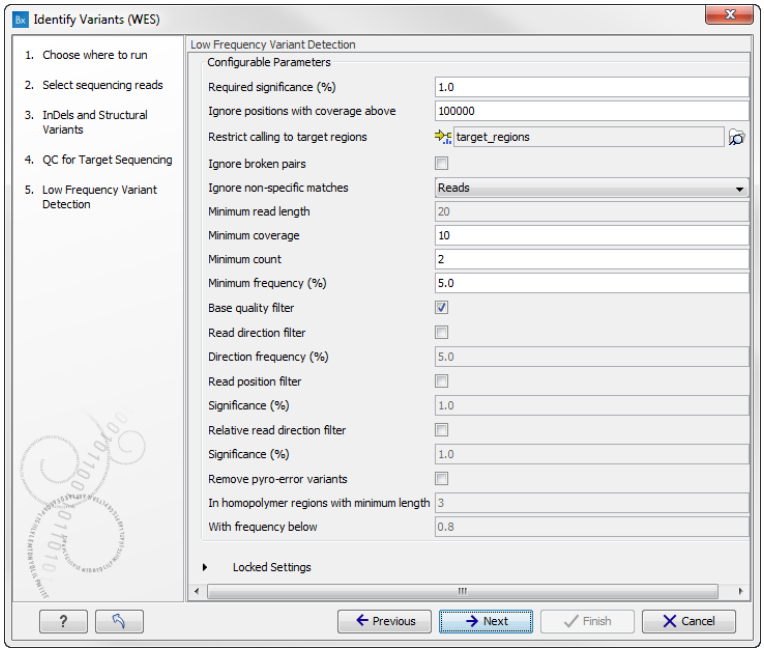


Figure 16.33: *Please specify the parameters for variant detection.*

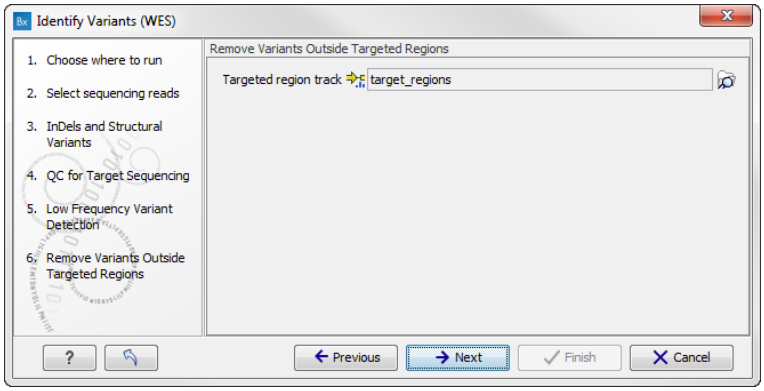


Figure 16.34: *Select the targeted region track. Variants found outside the targeted region will be removed.*

Output from the Identify Variants (WES) workflow

The **Identify Variants (WES)** tool produces six different types of output:

1. **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

2. **Target Regions Coverage** ( ) The target regions coverage track shows the coverage of the targeted regions. Detailed information about coverage and read count can be found in the table format, which can be opened by pressing the table icon found in the lower left corner of the **View Area**.

3. **Target Regions Coverage Report** ( ) The report consists of a number of tables and graphs that in different ways provide information about the targeted regions.

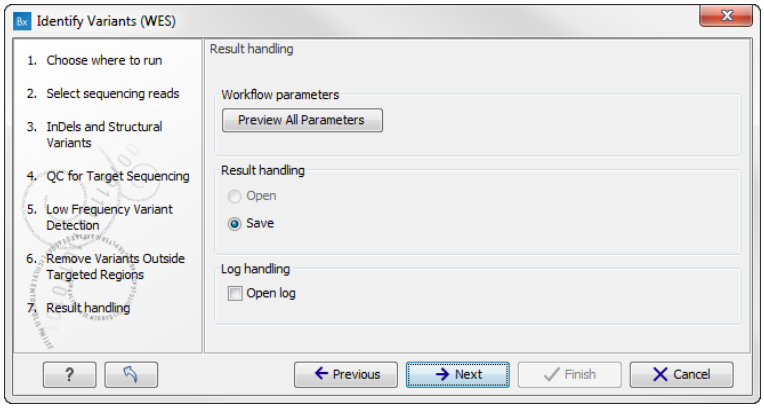


Figure 16.35: *Choose to save the results. In this wizard step you get the chance to preview the settings used in the ready-to-use workflow.*

4. **Identified Variants** ( ) A variant track holding the identified variants. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

5. **Genome Browser View Identify Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads, the identified variants, and the structural variants (see figure 16.5).

It is important that you do not delete any of the produced files individually as some of the outputs are linked to other outputs. If you would like to delete the outputs, please always delete all of them at the same time.

Please have first a look at the mapping report to see if the coverage is sufficient in regions of interest (e.g. > 30). Furthermore, please check that at least 90% of reads are mapped to the human reference sequence. In case of a targeted experiment, please also check that the majority of reads are mapping to the targeted region.

Afterwards please open the Genome Browser View file (see 16.36).

The Genome Browser View includes the track of identified variants in context to the human reference sequence, genes, transcripts, coding regions, targeted regions and mapped sequencing reads.

By double clicking on the variant track in the Genome Browser View, a table will be shown which includes information about all identified variants (see 16.37).

In case you like to change the reference sequence used for mapping as well as the human genes, please use the "Data Management".

16.2.4 Identify and Annotate Variants (WES)

The **Identify and Annotate Variants (WES)** tool should be used to identify and annotate variants in one sample. The tool consists of a workflow that is a combination of the **Identify Variants** and the **Annotate Variants** workflows.

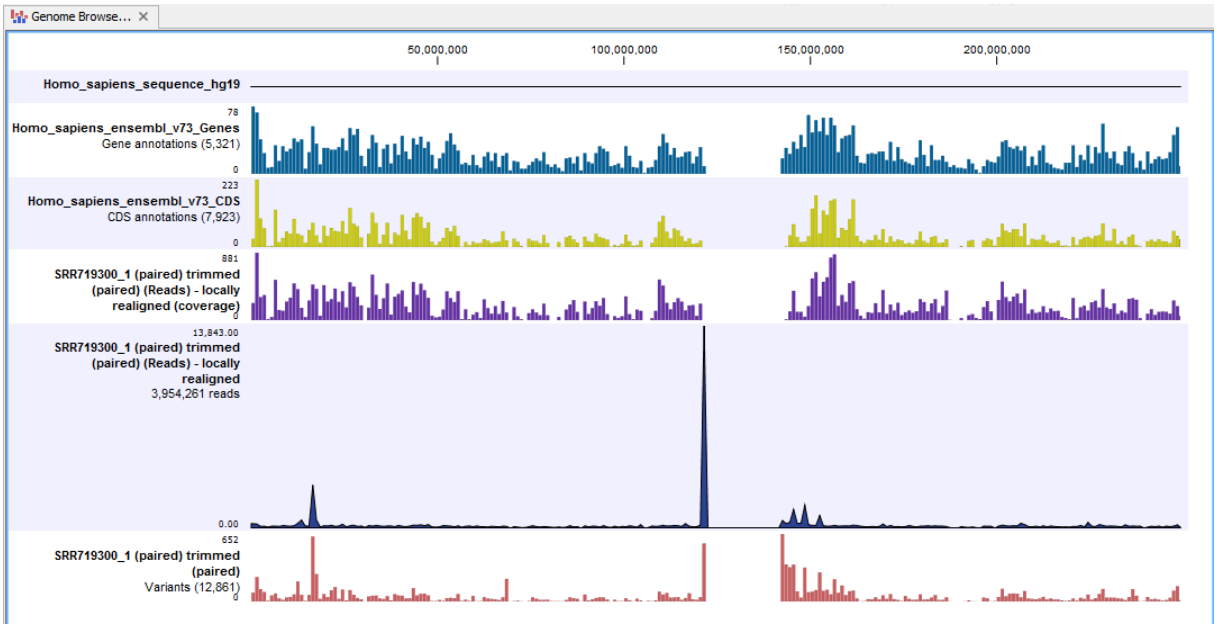


Figure 16.36: *The Genome Browser View allows you to inspect the identified variants in the context of the human genome.*

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection, which is run afterwards. After the variants have been detected, they are annotated with gene names, amino acid changes, conservation scores, information from relevant variants present in the ClinVar database, and information from common variants present in the common dbSNP, HapMap, and 1000 Genomes database. Furthermore, a detailed mapping report or a targeted region report (whole exome and targeted amplicon analysis) is created to inspect the overall coverage and mapping specificity.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit is available from the vendor of the enrichment kit and sequencing machine. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get the file in either .bed or .gff format.

To import the file:

Go to the toolbar | **Import (**) | **Tracks (**)

How to run the Identify and Annotate Variants (WES) workflow

To run the **Identify and Annotate Variants (WES)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Somatic Cancer** ( ) | **Identify and Annotate Variants (WES)** ( )

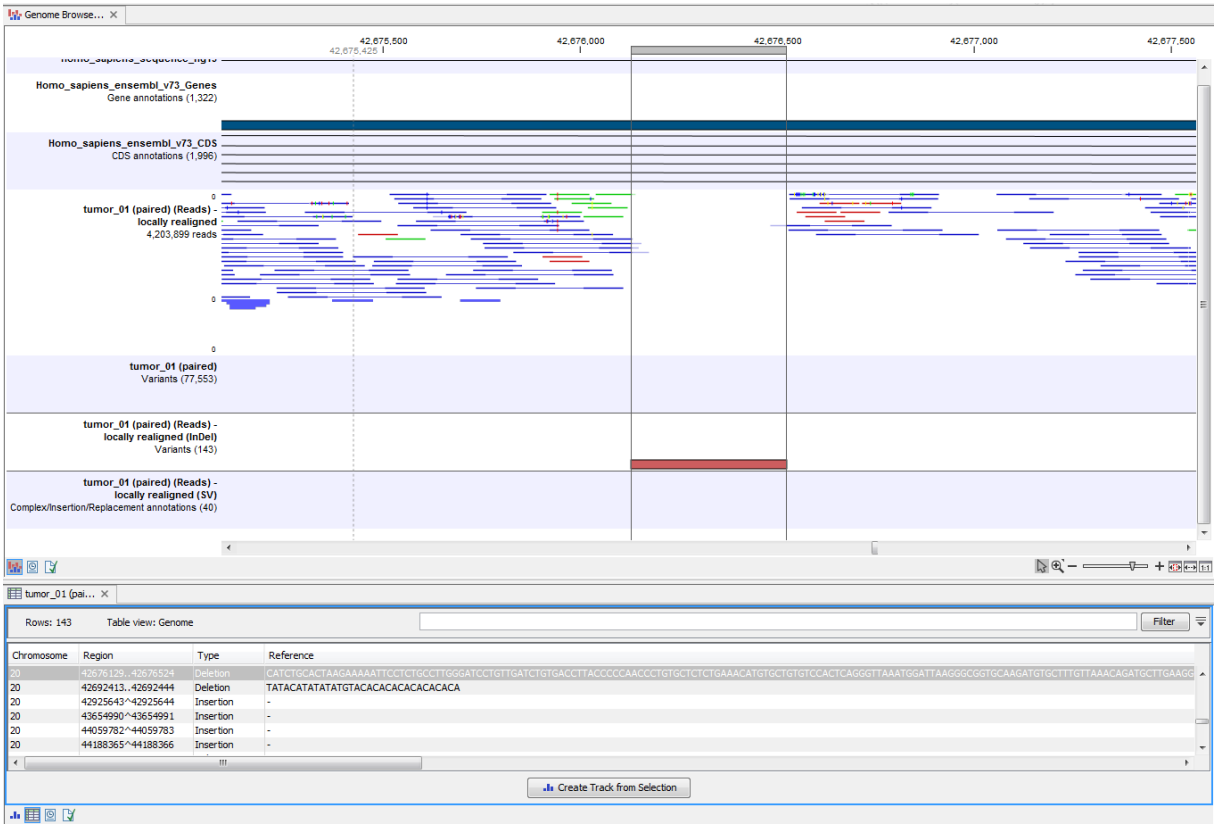


Figure 16.37: *Genome Browser View with an open track table to inspect identified variants more closely in the context of the human genome.*

1. Double-click on the **Identify and Annotate Variants (WES)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis. Click on the button labeled Next.

2. You can select the sequencing reads from the sample that should be analyzed.

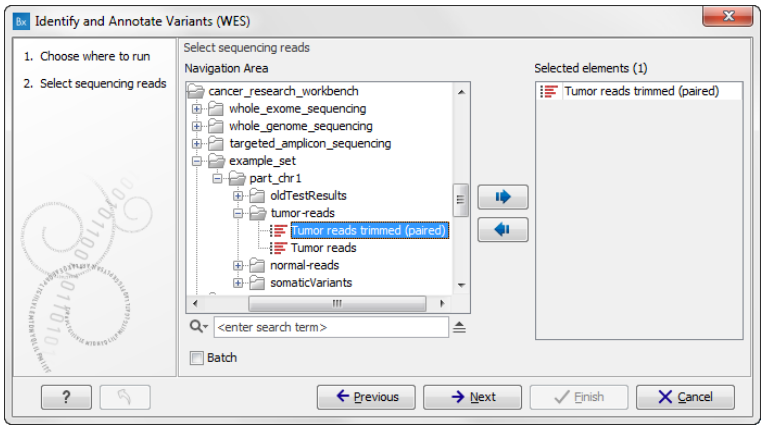


Figure 16.38: *Please select all sequencing reads from the sample to be analyzed.*

If several samples should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" (tick "Batch" at the bottom of the wizard as shown in figure 16.38)

and select the **folder** that holds the data you wish to analyse. If you have your sequencing data in separate folders, you should choose to run the analysis in batch mode.

When you have selected the sample(s) you wish to prepare, click on the button labeled **Next**.

3. In the next wizard step (figure 16.39) you can select the population from the 1000 Genomes project that you would like to use for annotation.

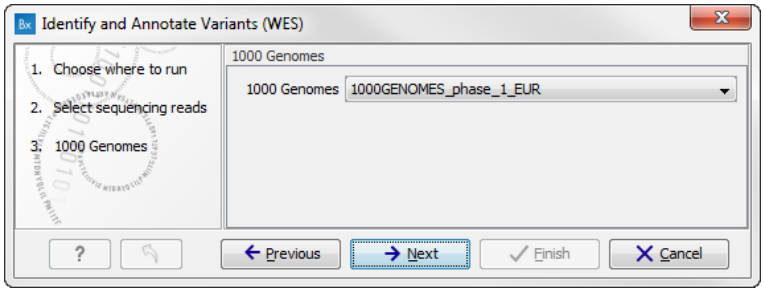


Figure 16.39: *Select the population from the 1000 Genomes project that you would like to use for annotation.*

4. In the "Indels and Structural Variants" dialog(figure 16.40), you can specify the target regions track. The variants found outside the targeted region will be removed at this step in the workflow.

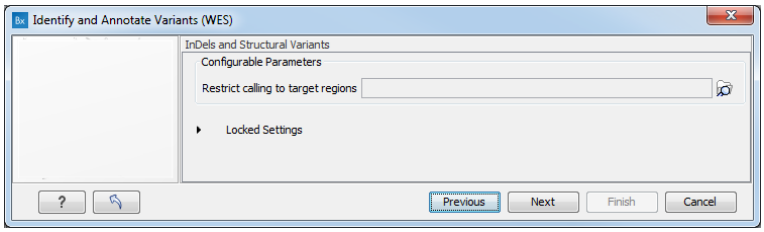


Figure 16.40: *In this wizard step you can specify the target regions track. Variants found outside these regions will be removed.*

5. In the next dialog (figure 16.41), you have to specify the parameters for the variant detection. For a description of the different parameters that can be adjusted, see section 22.14. If you click on "Locked Settings", you will be able to see all parameters used for variant detection in the ready-to-use workflow.

6. In the next wizard (figure 16.42) you can select the target region track and specify the minimum read coverage that should be present in the targeted regions.

7. Click on the button labeled **Next**, which will take you to the next wizard step (figure 16.43). Once again, select the relevant population from the 1000 Genomes project. This will add information from the 1000 Genomes project to your variants.

8. Click on the button labeled **Next**, which will take you to the next wizard step (figure 16.44). At this step you can select a population from the HapMap database. This will add information from the Hapmap database to your variants.

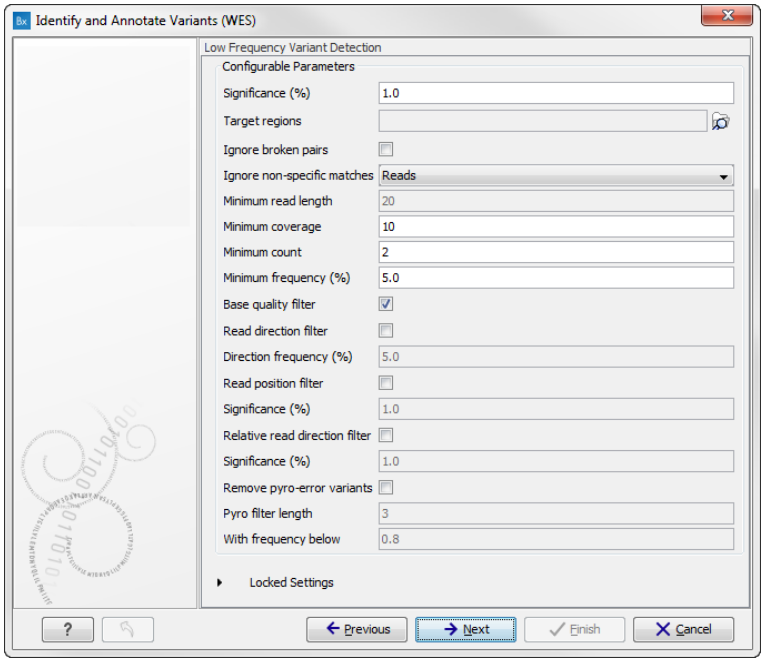


Figure 16.41: *Specify the parameters for variant calling.*

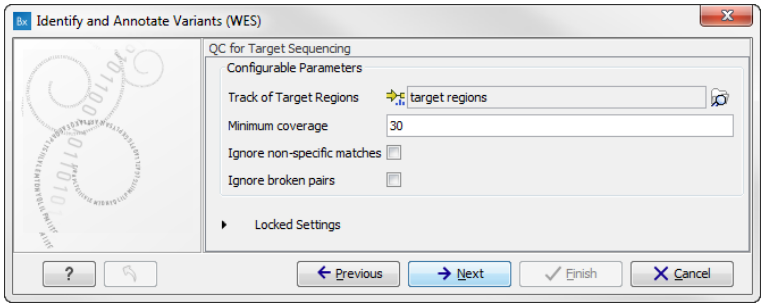


Figure 16.42: *Select the track with targeted regions from your experiment.*

9. In this wizard step (figure 16.45) you get the chance to check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

10. Choose to **Save** your results and press **Finish**.

    Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify and Annotate Variants (WES) workflow

The **Identify and Annotate Variants (WES)** tool produces several outputs.

Please do not delete any of the produced files alone as some of them are linked to other outputs. Please always delete all of them at the same time.

A good place to start is to take a look at the mapping report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Furthermore, please check that at least 90%

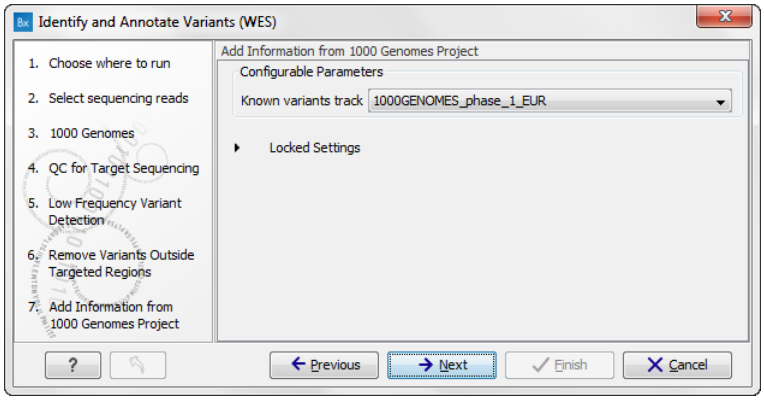


Figure 16.43: *Select the relevant population from the 1000 Genomes project. This will add information from the 1000 Genomes project to your variants.*

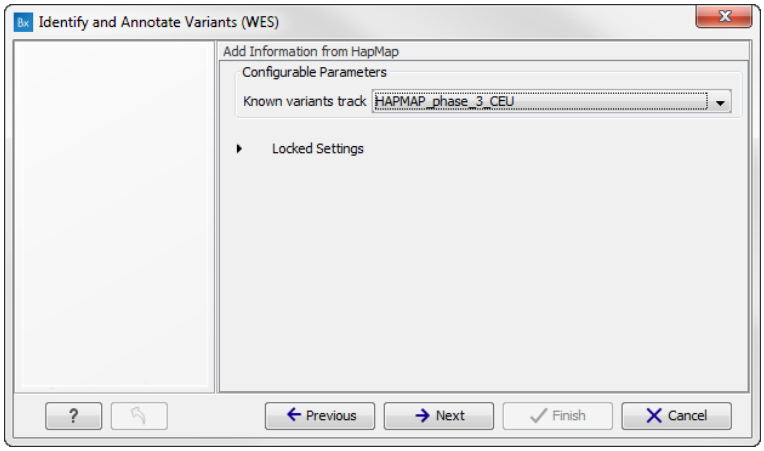


Figure 16.44: *Select a population from the HapMap database. This will add information from the Hapmap database to your variants.*

of the reads are mapped to the human reference sequence. In case of a targeted experiment, please also check that the majority of the reads are mapping to the targeted region.

Next, open the Genome Browser View file (see figure 16.46).

The Genome Browser View includes a track of the identified annotated variants in context to the human reference sequence, genes, transcripts, coding regions, targeted regions, mapped sequencing reads, relevant variants in the ClinVar database as well as common variants in common dbSNP, HapMap, and 1000 Genomes databases.

To see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant, a track with conservation scores is added as well.

By double-clicking on the annotated variant track in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see figure 16.47).

The added information will help you to identify candidate variants for further research. For example can common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) easily be seen.

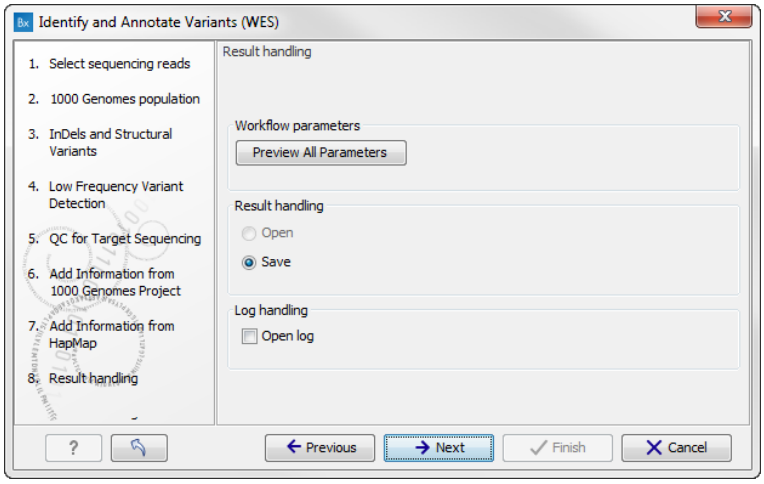


Figure 16.45: *Check the settings and save your results.*

Figure 16.46: *Genome Browser View to inspect identified variants in the context of the human genome and external databases.*

Not identified variants in ClinVar, can for example be prioritized based on amino acid changes (do they cause any changes on the amino acid level?). A high conservation level on the position of the variant between many vertebrates or mammals can also be a hint that this region could have an important functional role and variants with a conservation score of more than 0.9 (PhastCons score) should be prioritized higher. A further filtering of the variants based on their annotations can be facilitated using the table filter on top of the table.

If you wish to always apply the same filter criteria, the **Create new Filter Criteria** tool should be used to specify this filter and the **Identify and Annotate Variants (WES)** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion). See the reference manual for more information on how preinstalled workflows can be edited.

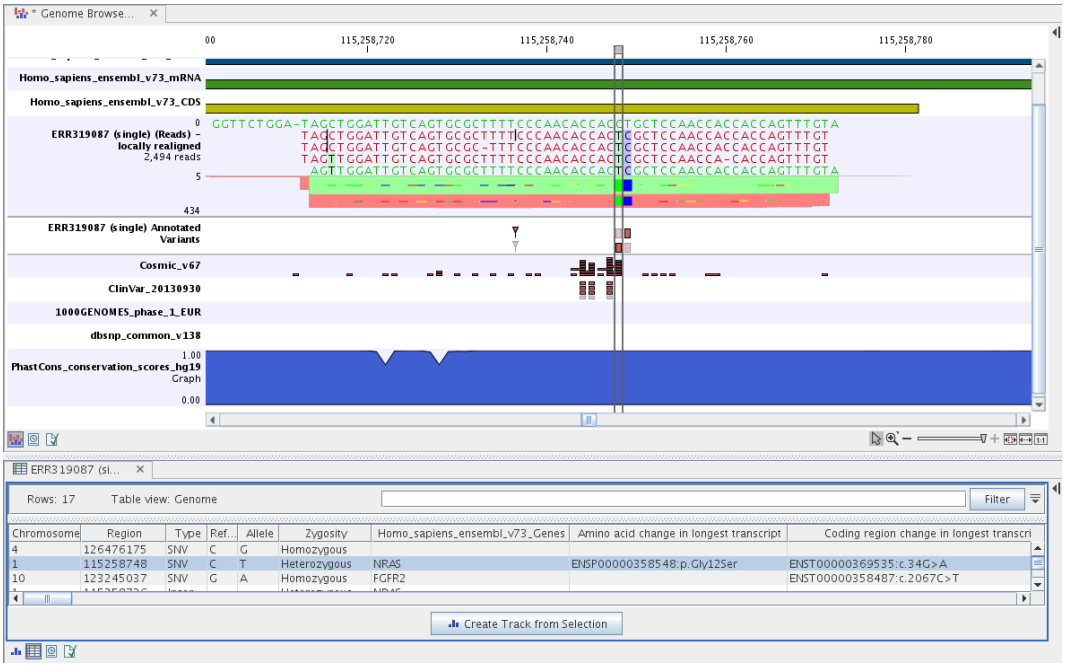


Figure 16.47: *Genome Browser View with an open track table to inspect identified somatic variants more closely in the context of the human genome and external databases.*

Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

In case you like to change the databases as well as the used database version, please use the "Data Management".

16.3 Hereditary Disease (WES)

16.3.1 Filter Causal Variants (WES-HD)

If you are analyzing a list of variants, you can use the **Filter Causal Variants (WES-HD)** ready-to-use workflow to remove variants that are outside the target region, as well as common variants present in publicly available databases. The workflow will annotate the remaining variants with gene names, conservation scores, and information from relevant databases.

The **Filter Causal Variants (WES-HD)** ready-to-use workflow accepts variants tracks files.

How to run the Filter Causal Variants (WES-HD) workflow

To run the **Filter Causal Variants (WES-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** () | **Hereditary Disease** () | **Filter Causal Variants (WES -HD)** ()

1. Double-click on the **Filter Somatic Variants (WES-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the **variant track** you want to use for filtering causal variants (figure 16.48).

The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the variant track name or click once on the file and then click on the arrow pointing to the right side in the middle of the wizard.

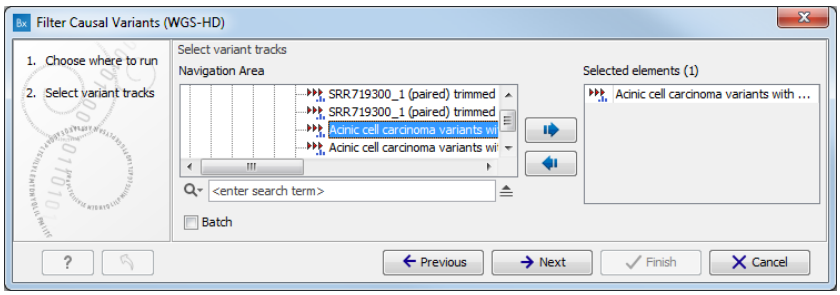


Figure 16.48: *Select the variant track from which you would like to filter somatic variants.*

3. Specify which of the **1000 Genomes populations** should be used for **annotation** (figure 16.49).

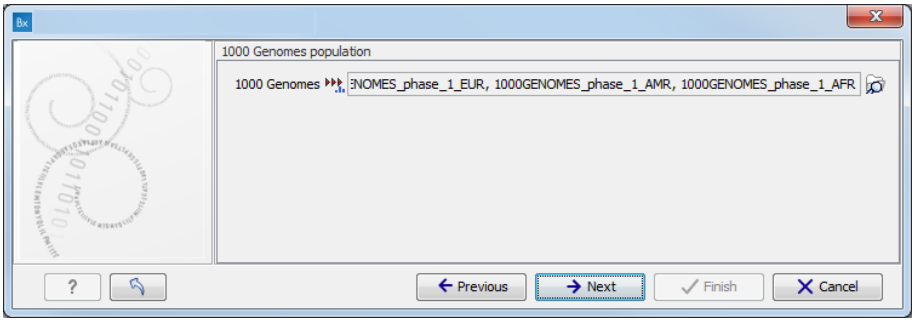


Figure 16.49: *Select the relevant 1000 Genomes population(s).*

4. Specify the **1000 Genomes population** that should be used for **filtering out** variants found in the 1000 Genomes project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

5. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 16.50).

   This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

6. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Filter Causal Variants (WES-HD) workflow

Three types of output are generated:

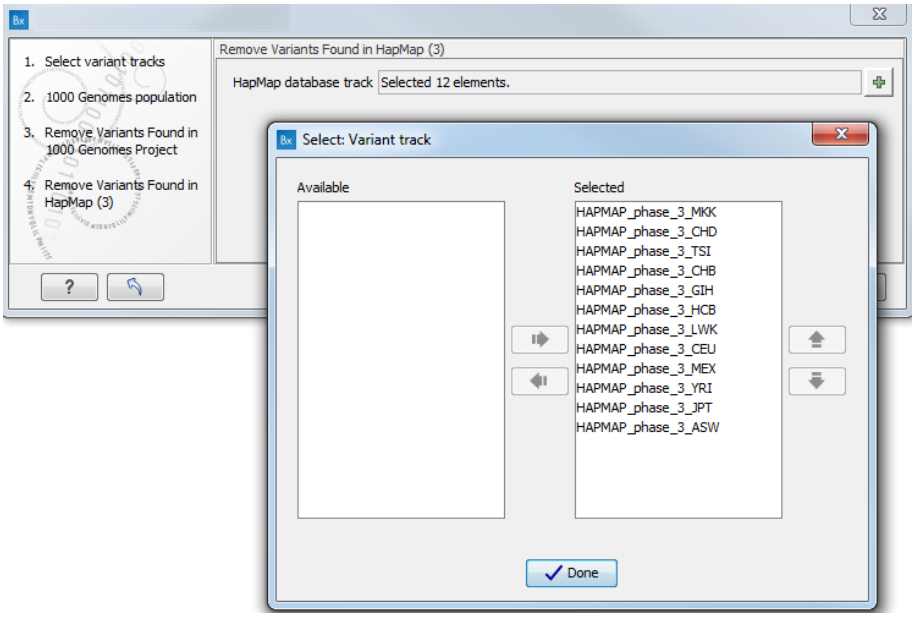


Figure 16.50: *Select the relevant Hapmap population(s).*

- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- A **Genome Browser View**
- A **Filtered Variant Track**

16.3.2 Identify Causal Inherited Variants in Family of Four (WES)

As the name of the workflow implies, you can use the **Identify Causal Inherited Variants in a Family of Four (WES)** ready-to-use workflow to identify inherited causal variants in a family of four. The family relationship can be a child, a mother, a father and one additional affected family member where, in addition to the child (the proband) one of the parents are affected and one additional family member is affected. The fourth family member can be any related and affected family member such as a sibling, grand parent, uncle or the like.

The **Identify Causal Inherited Variants in a Family of Four (WES)** ready-to-use workflow accepts sequencing reads as input from each of the four family members.

How to run the Identify Causal Inherited Variants in a Family of Four (WES) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Family of Four (WES)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Causal Inherited Variants in a Family of Four (WES)** ( )

1. Double-click on the **Identify Causal Inherited Variants in a Family of Four (WES)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 16.51).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

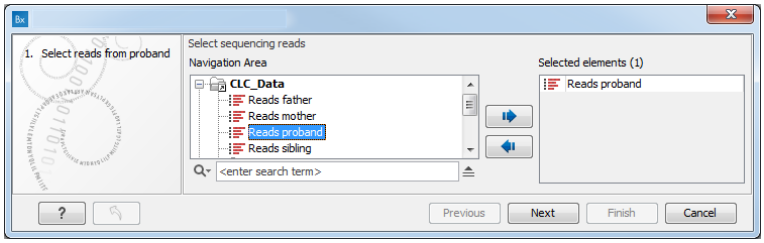


Figure 16.51: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **affected parent**.

4. Select the sequencing reads from the **unaffected parent**.

5. Select the sequencing reads from the **affected family member**.

6. Select the **targeted region** file (figure 16.52).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

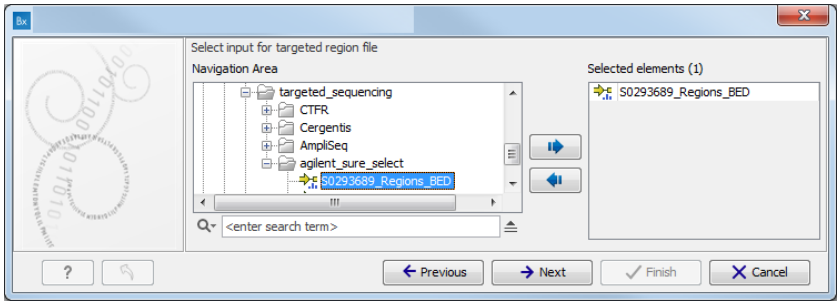


Figure 16.52: *Select the targeted region file you used for sequencing.*

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 16.53).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

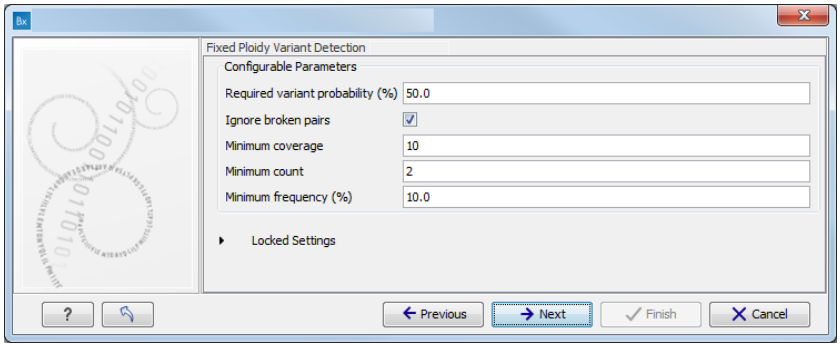


Figure 16.53: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
10. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected family member**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 16.54).

    When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

    The parameters that can be set are:

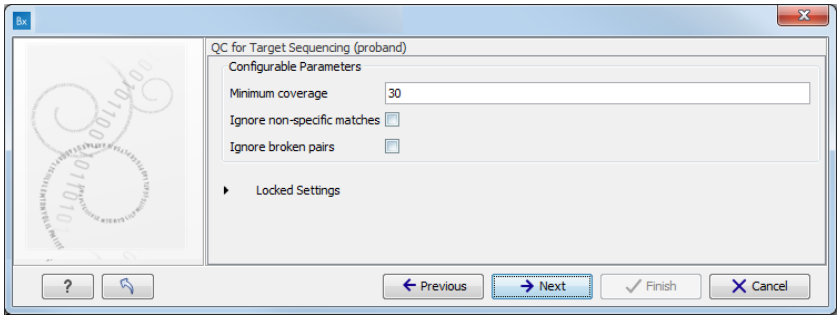


Figure 16.54: *Specify the parameters for the QC for Target Sequencing tool.*

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

12. Specify the parameters for the **QC for Target Sequencing** tool for the **affected parent**.
13. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected parent**.
14. Specify the parameters for the **QC for Target Sequencing** tool for the **affected family member**.
15. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 16.55).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

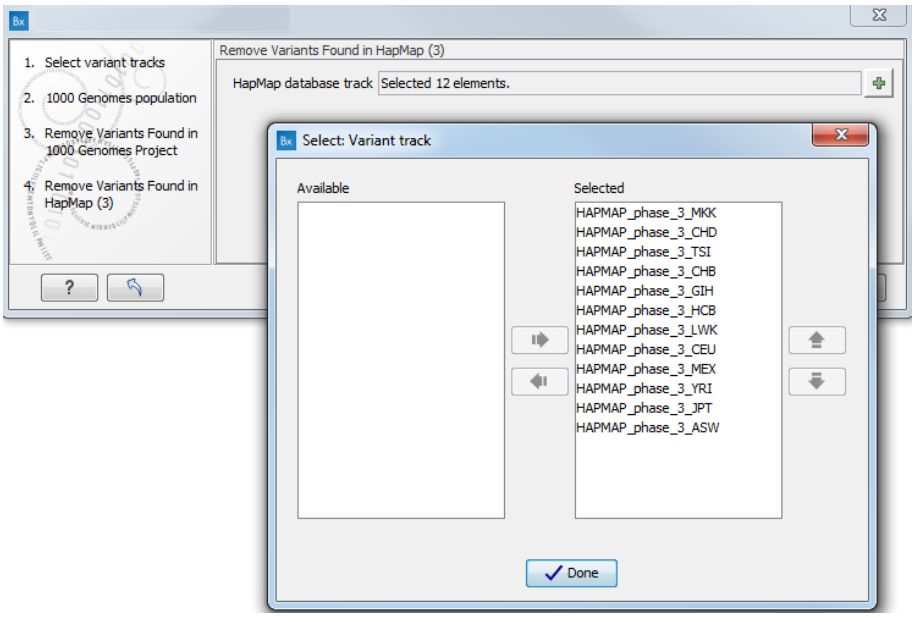


Figure 16.55: *Select the relevant Hapmap population(s).*

16. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Causal Inherited Variants in a Family of Four (WES) workflow

Six types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

16.3.3 Identify Causal Inherited Variants in Trio (WES)

The **Identify Causal Inherited Variants in a Trio (WES)** ready-to-use workflow identifies putative disease causing inherited variants by creating a list of variants present in both affected individuals and subtracting all variants in the unaffected individual. The workflow includes a back-check for all family members

The **Identify Causal Inherited Variants in a Trio (WES)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Causal Inherited Variants in a Trio (WES) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Trio (WES)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Causal Inherited Variants in a Trio (WES)** ( )

1. Double-click on the **Identify Causal Inherited Variants in a Trio (WES)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 16.56).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

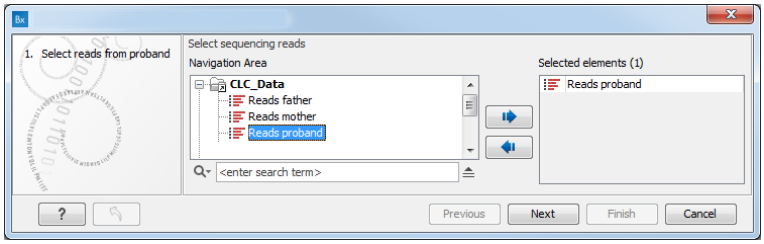


Figure 16.56: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads for the **affected parent**.

4. Select the sequencing reads for the **unaffected parent**.

5. Select the targeted region file (figure 16.57).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

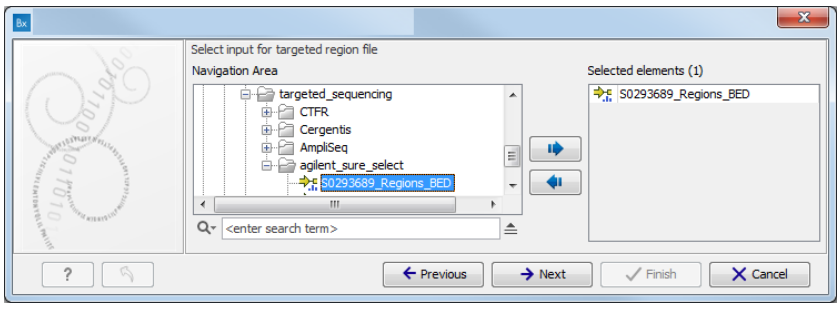


Figure 16.57: *Select the targeted region file you used for sequencing.*

6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 16.58).

The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

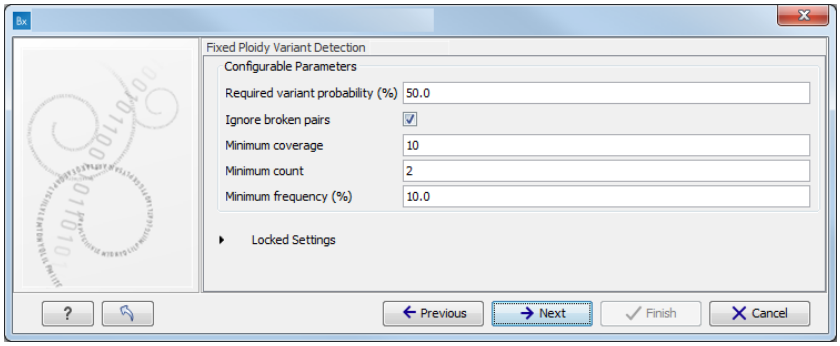


Figure 16.58: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
9. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 16.59).

When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

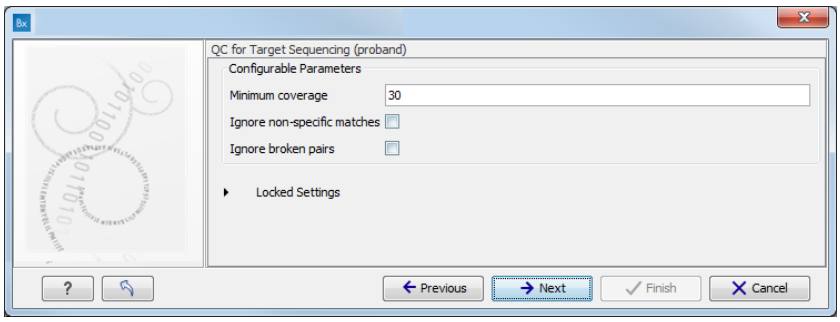


Figure 16.59: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

10. Specify the parameters for the **QC for Target Sequencing** tool for the **affected parent**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected parent**.
12. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 16.60).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).
13. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Causal Inherited Variants in a Trio (WES) workflow

Six types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.

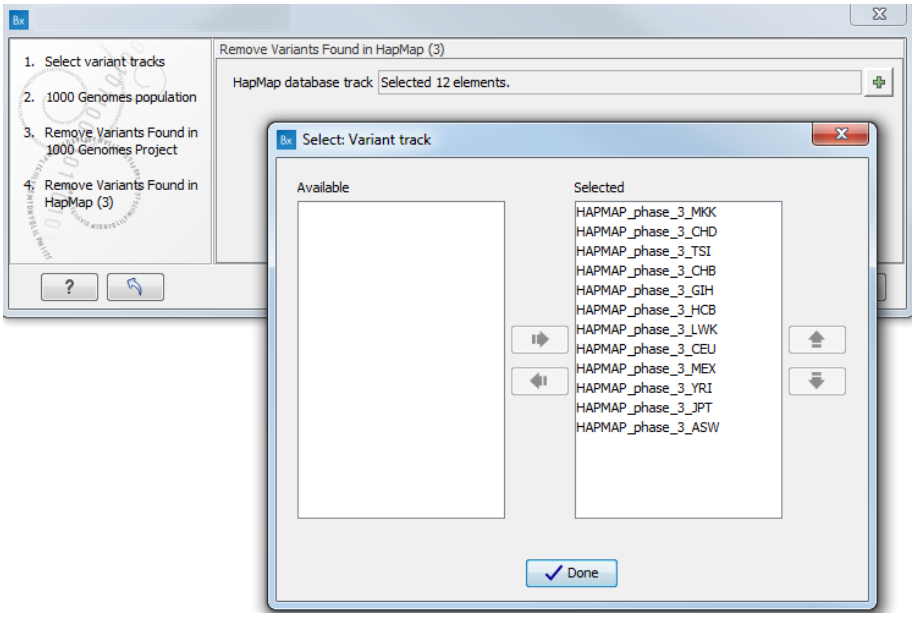


Figure 16.60: *Select the relevant Hapmap population(s).*

- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

16.3.4 Identify Rare Disease Causing Mutations in Family of Four (WES)

You can use the **Identify Rare Disease Causing Mutations in a Family of Four (WES)** ready-to-use workflow to identifie de novo and compound heterozygous variants from an extended family of four, where the fourth individual is not affected.

The **Identify Rare Disease Causing Mutations in a Family of Four (WES)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Family of Four (WES) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Family of Four (WES)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** | **Hereditary Disease** | **Identify Rare Disease Causing Mutations in a Family of Four (WGS**

1. Double-click on the **Identify Rare Disease Causing Mutations in a Family of Four (WES)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 16.61).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

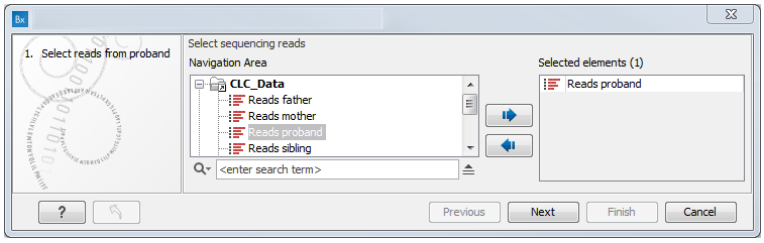


Figure 16.61: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **mother**.
4. Select the sequencing reads from the **father**.
5. Select the sequencing reads from for the **unaffected sibling**.
6. Select the **targeted region** file (figure 16.62).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.
7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 16.63).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

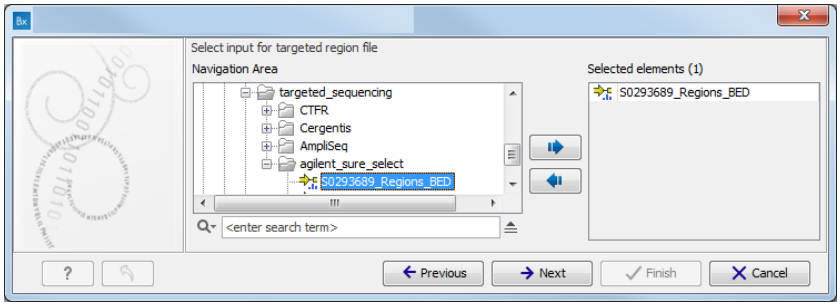


Figure 16.62: *Select the targeted region file you used for sequencing.*

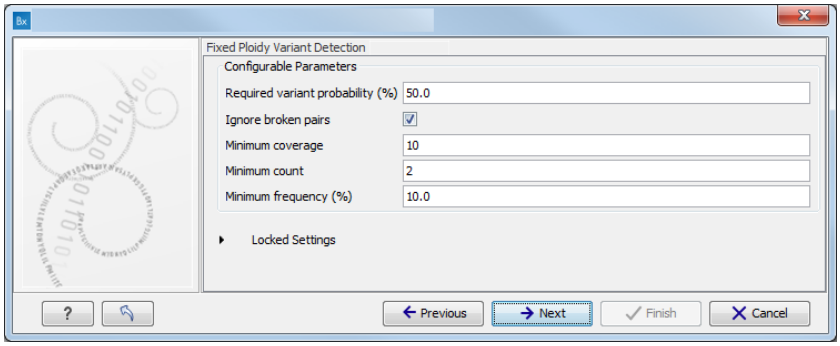


Figure 16.63: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **mother**.

9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **father**.
10. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **sibling**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 16.64).

    When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

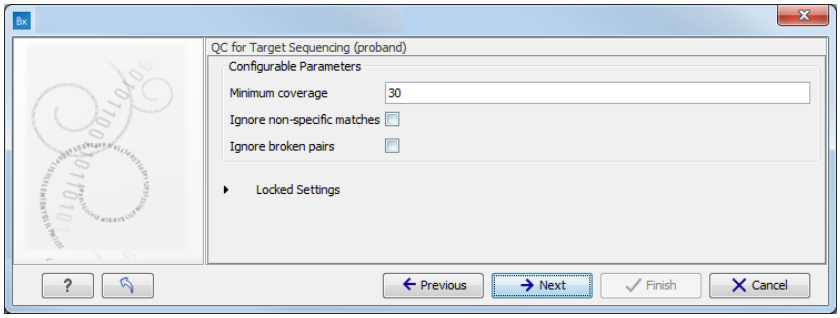


Figure 16.64: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

12. Specify the parameters for the **QC for Target Sequencing** tool for the **mother**.
13. Specify the parameters for the **QC for Target Sequencing** tool for the **father**.
14. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected sibling**.
15. Specify the affected child's **gender** (figure 16.65).

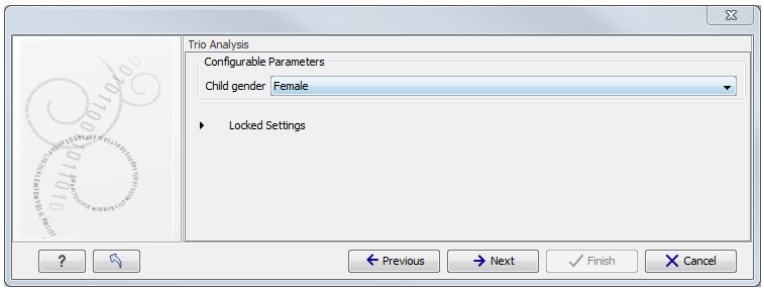


Figure 16.65: *Specify the proband's gender.*

16. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **mother** (figure 16.66).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

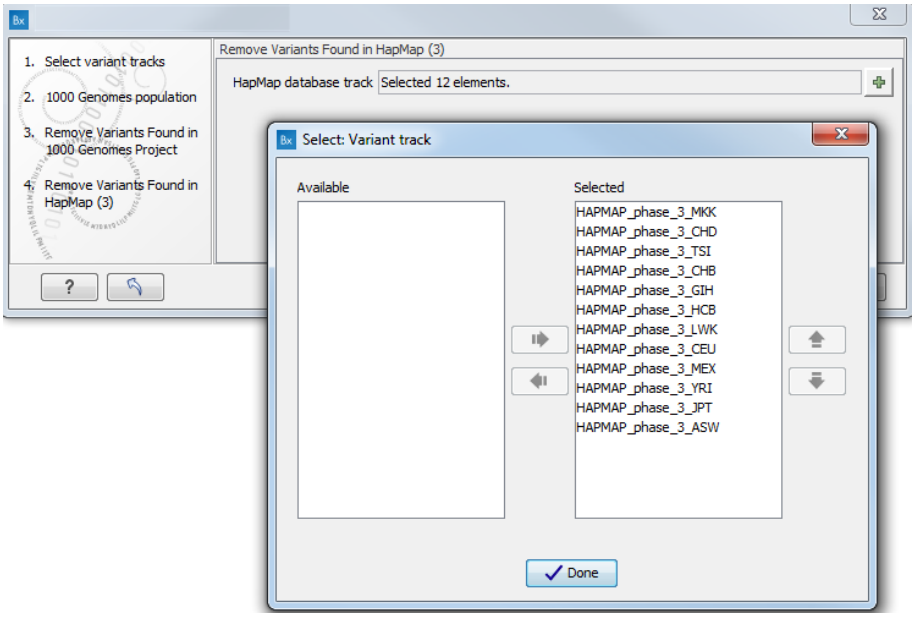


Figure 16.66: *Select the relevant Hapmap population(s).*

17. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father**.

18. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.

19. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.

20. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Rare Disease Causing Mutations in a Family of Four (WES) workflow

Twelve different types of output are generated:

- **Reads Mapping** One for each family member. The reads mapped to the reference sequence.
- **Variant Tracks** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.

- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with de novo Variants** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.
- **De novo variants** Variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

16.3.5 Identify Rare Disease Causing Mutations in Trio (WES)

The **Identify Rare Disease Causing Mutations in a Trio (WES)** identifies de novo and compound heterozygous variants from a Trio. The workflow includes a back-check for all family members.

The **Identify Rare Disease Causing Mutations in a Trio (WES)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Trio (WES) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Trio (WES)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** () | **Hereditary Disease** () | **Identify Rare Disease Causing Mutations in a Trio (WES** ()

1. Double-click on the **Identify Rare Disease Causing Mutations in a Trio (WES)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 16.67).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

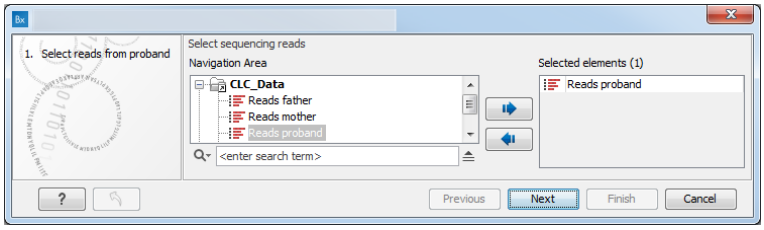


Figure 16.67: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **mother**.

4. Select the sequencing reads from the **father**.

5. Select the **targeted region** file (figure 16.68).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

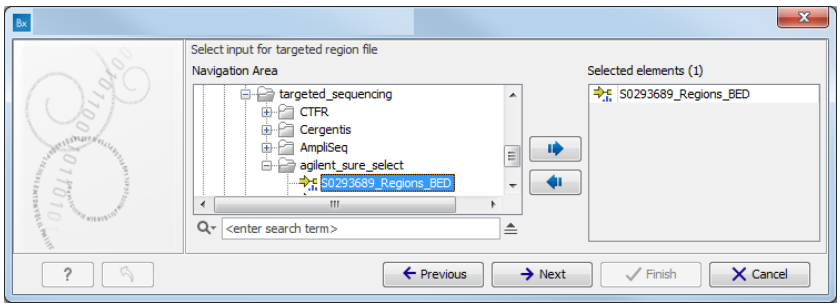


Figure 16.68: *Select the targeted region file you used for sequencing.*

6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 16.69).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

   - **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

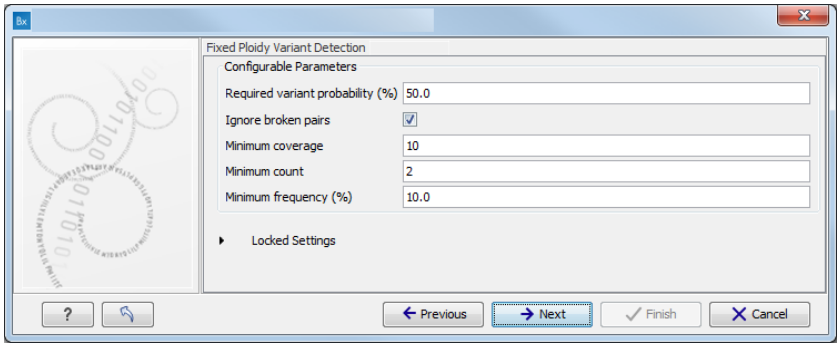


Figure 16.69: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **mother**.
8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **father**.
9. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 16.70).

   When working with targeted data, quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

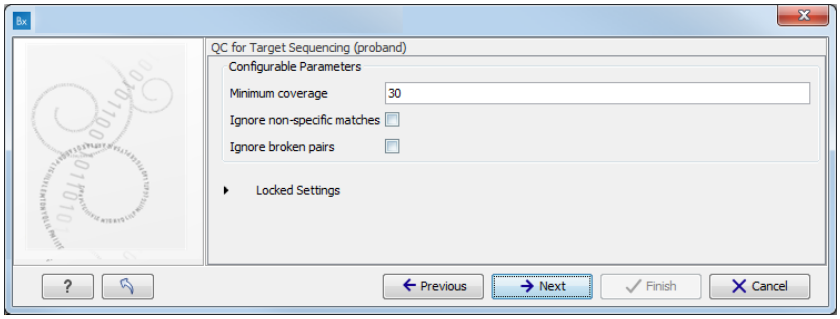


Figure 16.70: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

10. Specify the parameters for the **QC for Target Sequencing** tool for the **mother**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **father**.
12. Specify the affected child's **gender** (figure 16.71).

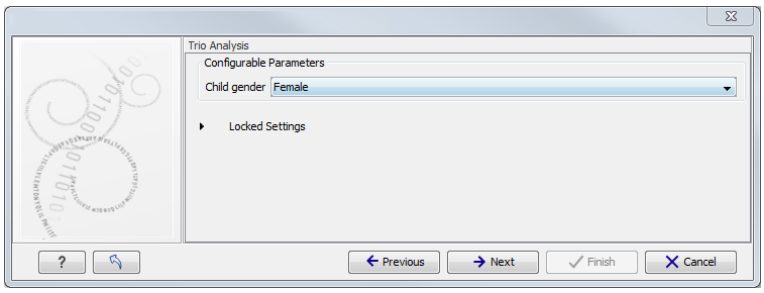


Figure 16.71: *Specify the proband's gender.*

13. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **mother** (figure 16.72).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

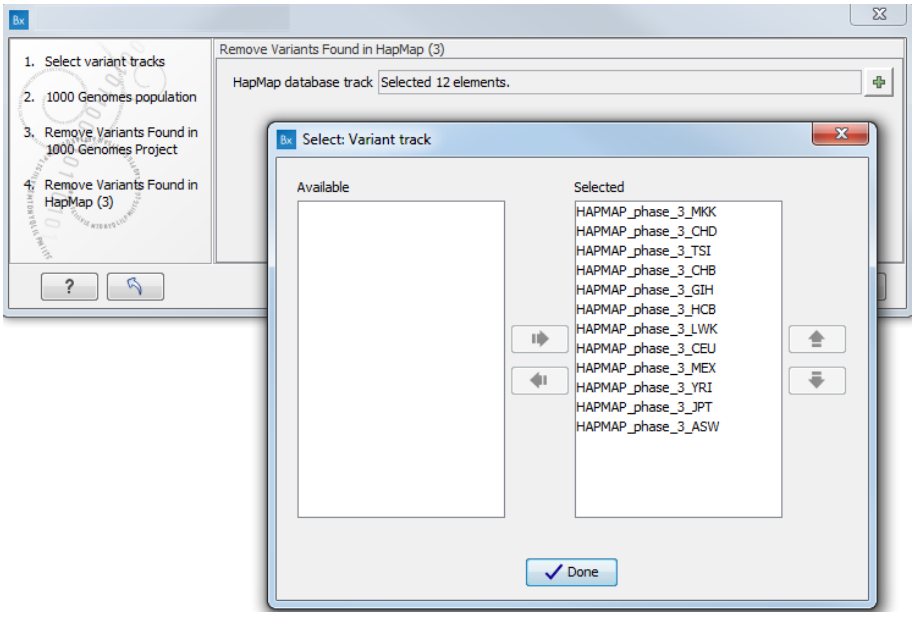


Figure 16.72: *Select the relevant Hapmap population(s).*

14. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father**.

15. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.
16. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.
17. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Rare Disease Causing Mutations in a Trio (WES) workflow

Twelve different types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variant Tracks** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **De novo variants** Variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with de novo Variants** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**

16.3.6 Identify Variants (WES-HD)

You can use the **Identify Variants (WES-HD)** ready-to-use workflow to call variants in the mapped and locally realigned reads. The workflow removes false positives and, in case of a targeted experiment, removes variants outside the targeted region. Variant calling is performed with the Fixed Ploidy Variant Detection tool.

The **Identify Variants (WES-HD)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Variants (WES-HD) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Variants (WES-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** ( ) | **Hereditary Disease** ( ) | **Identify Variants (WES-HD)** ( )

1. Double-click on the **Identify Variants (WES-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads you want to analyze (figure 16.73). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

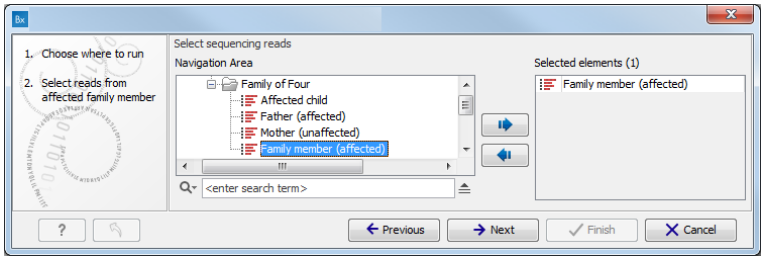


Figure 16.73: *Specify the sequencing reads for the appropriate family member.*

3. Specify a target region file for the **Indels and Structural Variants** tool (figure 16.74).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

   item Specify the parameters for the **Fixed Ploidy Variant Detection** tool (figure 16.75).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

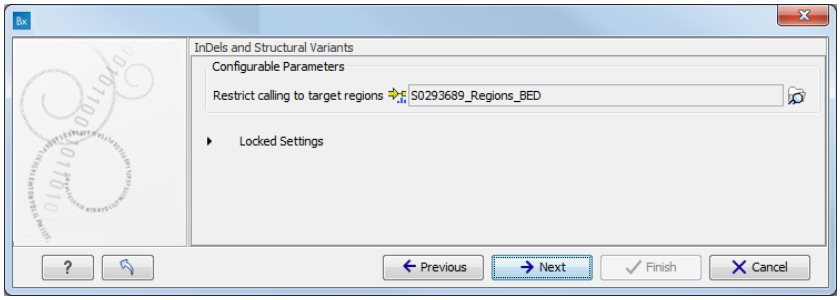


Figure 16.74: *Specify the parameters for the Indels and Structural Variants tool.*

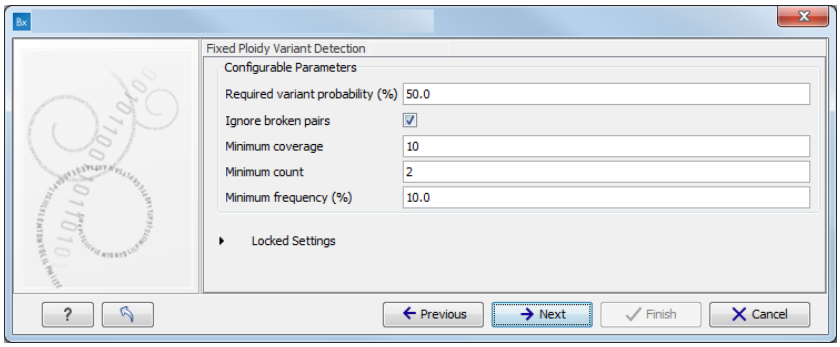


Figure 16.75: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

4. Specify the parameters for the **QC for Target Sequencing** tool, including a Target region file (figure 16.76).

When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

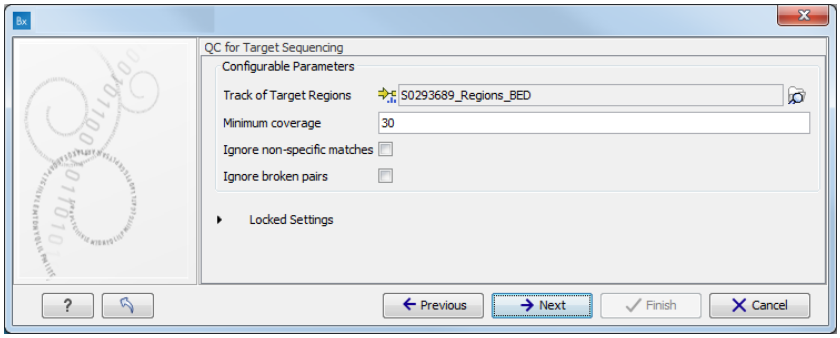


Figure 16.76: *Specify the parameters for the QC for Target Sequencing tool.*

For more information about the tool, see section 20.1.

5. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Variants (WES-HD) workflow

Four types of output are generated:

- A **Reads Track** Read Mapping
- A **Filtered Variant Track** Identified variants
- A **Coverage Report**
- A **Per-region Statistics Track**

16.3.7 Identify and Annotate Variants (WES-HD)

The **Identify and Annotate Variants (WES-HD)** tool should be used to identify and annotate variants in one sample. The tool consists of a workflow that is a combination of the **Identify Variants** and the **Annotate Variants** workflows.

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection, which is run afterwards. After the variants have been detected, they are annotated with gene names, amino acid changes, conservation scores, information from relevant variants present in

the ClinVar database, and information from common variants present in the common dbSNP, HapMap, and 1000 Genomes database. Furthermore, a targeted region report is created to inspect the overall coverage and mapping specificity.

How to run the Identify and Annotate Variants (WES-HD) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify and Annotate Variants (WES-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Exome Sequencing** () | **Hereditary Disease** () | **Identify and Annotate Variants (WES-HD)** ()

1. Double-click on the **Identify and Annotate Variants (WES-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads you want to analyze (figure 16.77). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

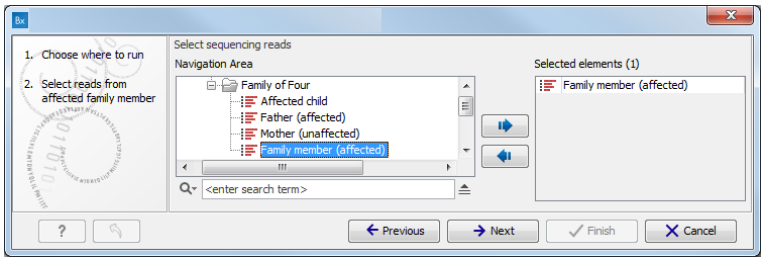


Figure 16.77: *Specify the sequencing reads for the appropriate family member.*

3. Specify which 1000 Genomes population you would like to use (figure 16.78).

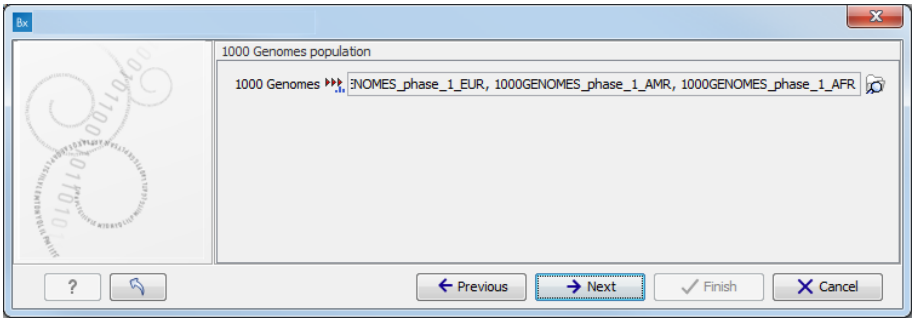


Figure 16.78: *Select the relevant 1000 Genomes population(s).*

4. Specify a target region file for the **Indels and Structural Variants** tool. (figure 16.79).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is

something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

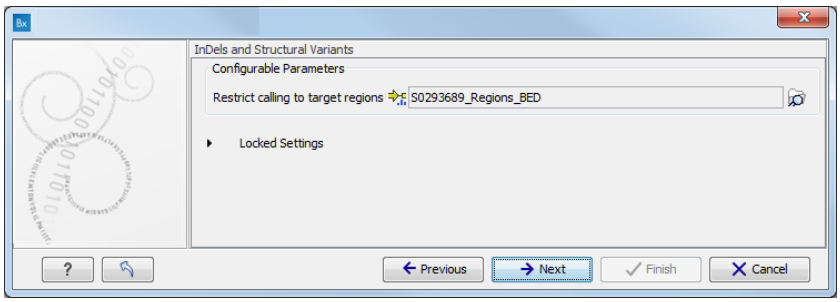


Figure 16.79: *Specify the parameters for the Indels and Structural Variants tool.*

5. Specify the **Fixed Ploidy Variant Detection** settings, including a target region file (figure 16.80).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

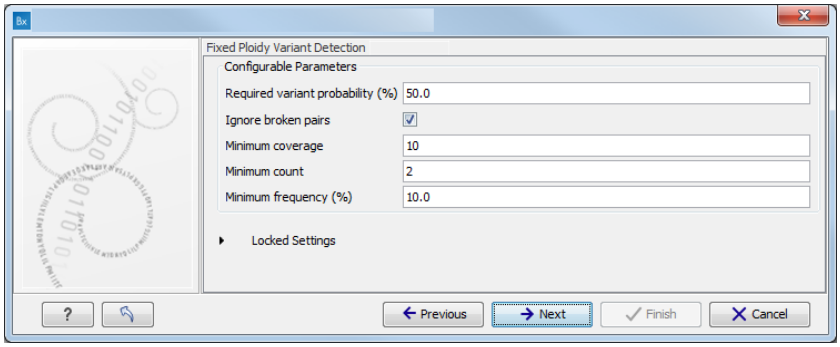


Figure 16.80: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may

also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.

- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

6. Specify the parameters for the **QC for Target Sequencing** tool, including a target region file (figure 16.81).

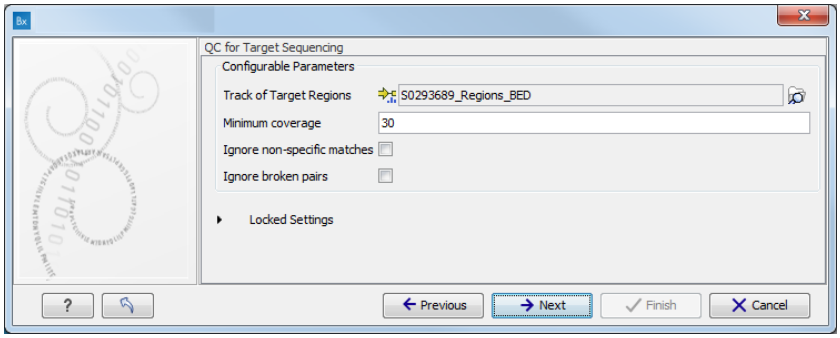


Figure 16.81: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

For more information about the tool, see section 20.1.

7. Specify a **targeted region** file to remove variants outside of this region. (figure 16.82)

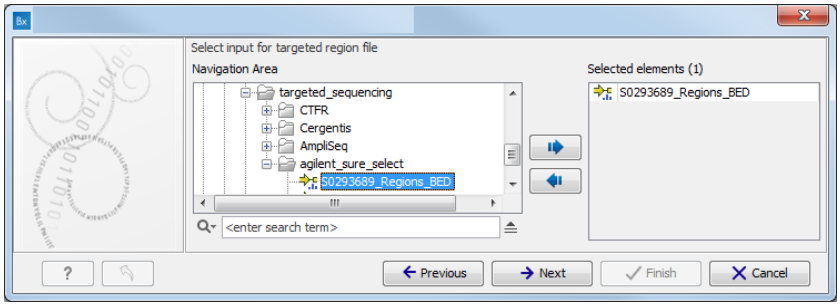


Figure 16.82: *Select the targeted region file you used for sequencing.*

8. Specify the 1000 Genomes population that should be used to add information on variants found in the 1000 Genomes project. This can be done using the drop-down list found in

this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

9. Specify the Hapmap population that should be used to add information on variants found in the Hapmap project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

10. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify and Annotate Variants (WES-HD) workflow

Six types of output are generated:

- A **1 Reads Track**
- A **1 Coverage Report Read Mapping**
- A **1 Per-region Statistics Track**
- A **Filtered Variant Track** Annotated variants
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- A **Genome Browser View**

Chapter 17

Targeted amplicon sequencing (TAS)

Contents

Targeted sequencing, also known as "targeted resequencing" or "amplicon sequencing" is a focused approach to genome sequencing with only selected areas of the genome being sequenced. In cancer research and diagnostics, targeted sequencing is usually based on sequencing panels that target a number of known cancer-associated genes.

Thirteen ready-to-use workflows are available for analysis of targeted amplicon sequencing data (figure 17.1). The concept of the pre-installed ready-to-use workflows is that read data are used as input in one end of the workflow and in the other end of the workflow you get a track based genome browser view and a table with all the identified variants, which may or may not have been subjected to different kinds of filtering and/or annotation.

In this chapter we will discuss what the individual ready-to-use workflows can be used for and go through step by step how to run the workflows.

Note! Often you will have to prepare data with one of the two **Preparing Raw Data** workflows described in section 14 before you proceed to **Automatic analysis of sequencing data (TAS)**.

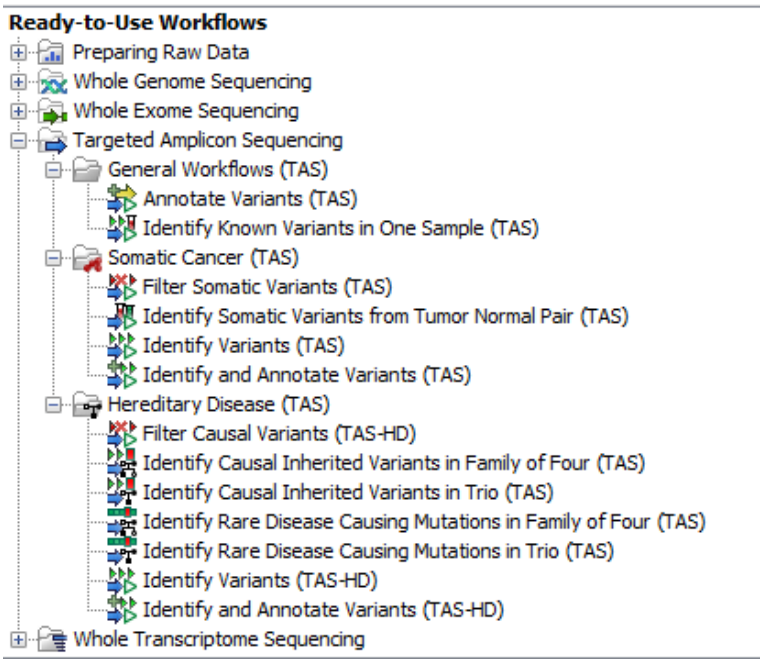


Figure 17.1: *The eleven workflows available for analyzing targeted amplicon sequencing data.*

17.1 General Workflows (TAS)

17.1.1 Annotate Variants (TAS)

Using a variant track () (e.g. the output from the Identify Variants ready-to-use workflow) the **Annotate Variants (TAS)** ready-to-use workflow runs an "internal" workflow that adds the following annotations to the variant track:

- **Gene names** Adds names of genes whenever a variant is found within a known gene.
- **mRNA** Adds names of mRNA whenever a variant is found within a known transcript.
- **CDS** Adds names of CDS whenever a variant is found within a coding sequence.
- **Amino acid changes** Adds information about amino acid changes caused by the variants.
- **Information from ClinVar** Adds information about the relationships between human variations and their clinical significance.
- **Information from dbSNP** Adds information from the "Single Nucleotide Polymorphism Database", which is a general catalog of genome variation, including SNPs, multinucleotide polymorphisms (MNPs), insertions and deletions (indels), and short tandem repeats (STRs).
- **PhastCons Conservation scores** The conservation scores, in this case generated from a multiple alignment with a number of vertebrates, describe the level of nucleotide conservation in the region around each variant.

How to run the Annotate Variants (TAS) workflow

1. Go to the toolbox and select the **Annotate Variants (TAS)** workflow. In the first wizard step, select the input variant track (figure 17.2).

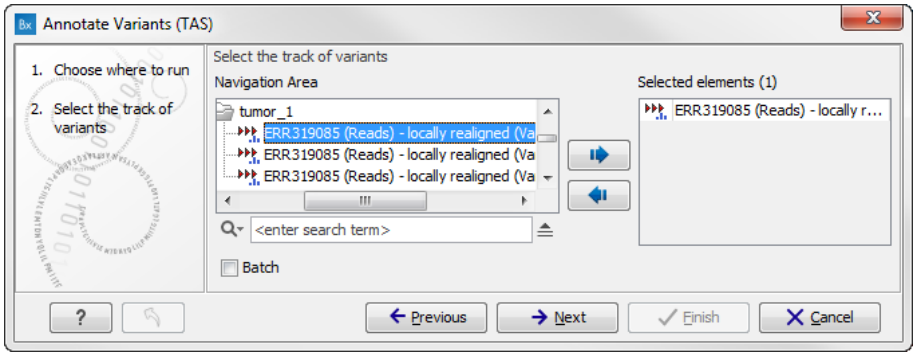


Figure 17.2: *Select the variant track to annotate.*

2. Click on the button labeled **Next**. The only parameter that should be specified by the user is which 1000 Genomes population you use (figure 17.3). This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

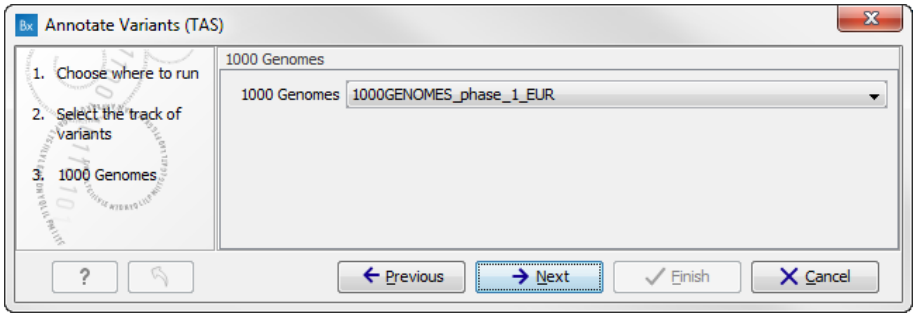


Figure 17.3: *Select the relevant 1000 Genomes population(s).*

3. Click on the button labeled **Next** to go to the last wizard step (figure 17.4).

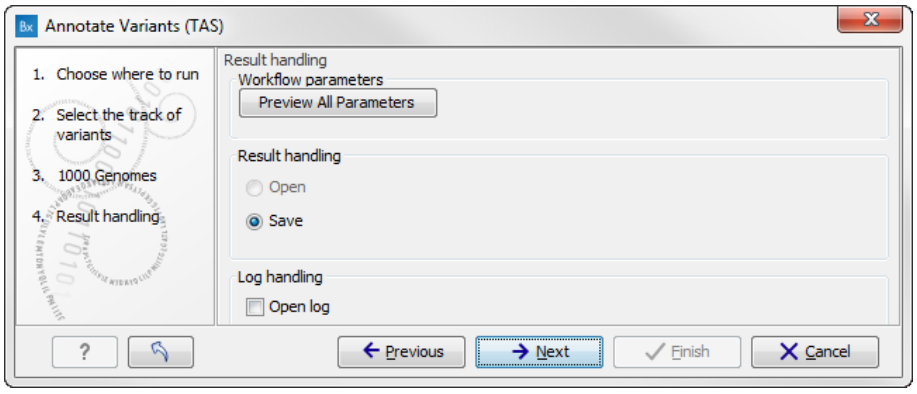


Figure 17.4: *Check the settings and save your results.*

In this wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

4. Choose to **Save** your results and click on the button labeled **Finish**.

Output from the Annotate Variants (TAS) workflow

Two types of output are generated:

1. **Annotated Variants** ( ) Annotation track showing the variants. Hold the mouse over one of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant.
2. An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
3. **Genome Browser View Annotated Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in dbSNP, ClinVar, 1000 Genomes, and PhastCons conservation scores (see figure 17.5).

Note! Please be aware, that if you delete the annotated variant track, this track will also disappear from the genome browser view.

It is possible to add tracks to the Genome Browser View such as mapped sequencing reads as well as other tracks. This can be done by dragging the track directly from the **Navigation Area** to the Genome Browser View.

If you double-click on the name of the annotated variant track in the left hand side of the Genome Browser View, a table that includes all variants and the added information/annotations will open (see figure 17.6). The table and the Genome Browser View are linked; if you click on an entry in the table, this particular position in the genome will automatically be brought into focus in the Genome Browser View.

You may be met with a warning as shown in figure 17.7. This is simply a warning telling you that it may take some time to create the table if you are working with tracks containing large amounts of annotations. Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar database, can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments

Figure 17.5: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list) containing individual tracks for all added annotations.*

where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** ( ) | **Create Filter Criteria** ( )

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

17.1.2 Identify Known Variants in One Sample (TAS)

The **Identify Known Variants in One Sample (TAS** ready-to-use workflow is a combined data analysis and interpretation ready-to-use workflow.

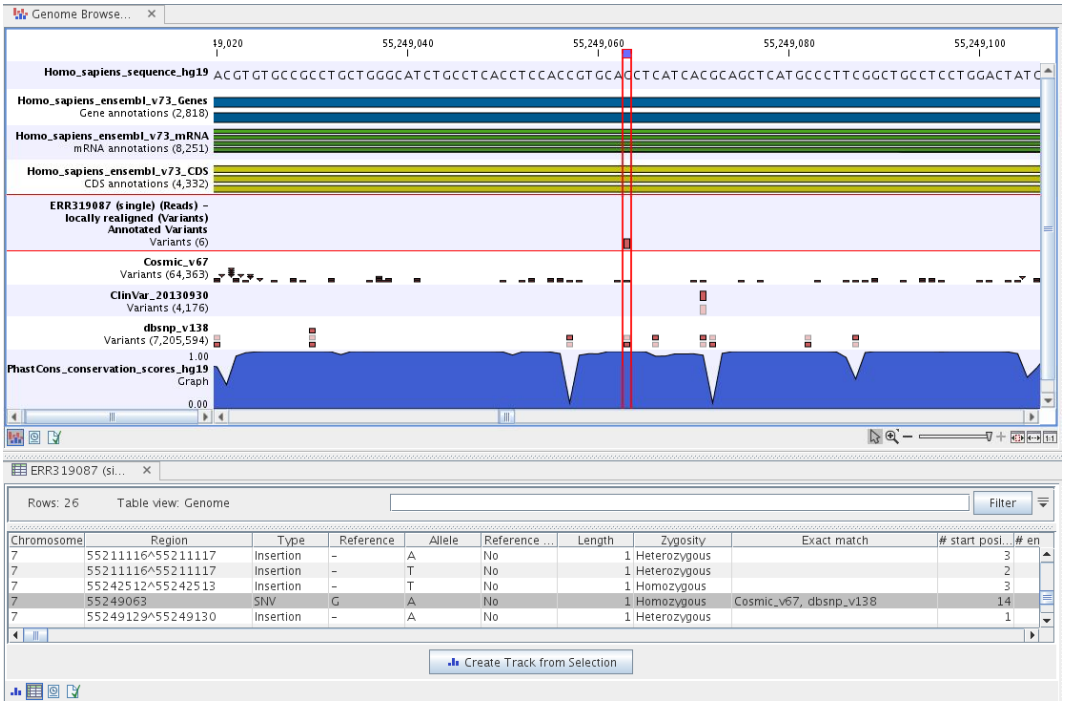


Figure 17.6: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list). The information is also available in table view. Click on the small table icon to open the table view. If you hold down the "Ctrl" key while clicking on the table icon, you will open a split view showing both the genome browser view and the table view.*

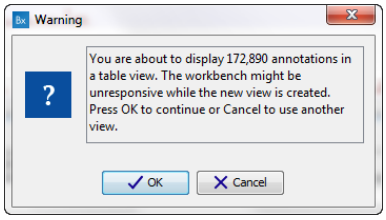


Figure 17.7: *Warning that appears when you work with tracks containing many annotations.*

It should be used to identify known variants, specified by the user (e.g. known breast cancer associated variants), for their presence or absence in a sample.

Please note that the ready-to-use workflow will not identify new variants.

The **Identify Known Variants in One Sample (TAS)** ready-to-use workflow maps the sequencing reads to a human genome sequence and does a local realignment of the mapped reads to improve the subsequent variant detection. In the next step, only variants specified by the user are identified and annotated in the newly generated read mapping.

Import your known variants

To make an import into the *Biomedical Genomics Workbench*, you should have your variants in GVF format (http://www.sequenceontology.org/resources/gvf.html or VCF format http://ga4gh.org/#/fileformats-team).

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit will be provided by the vendor. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get it in either .bed or .gff format.

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

How to run the Identify Known Variants in One Sample (TAS) workflow

1. Go to the toolbox and double-click on

   Toolbox | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **General Workflows (TAS)** | **Identify Known Variants from One Sample (TAS)** ()

2. This will open the wizard step shown in figure 17.8 where you can select the reads of the sample, which should be tested for presence or absence of your known variants.

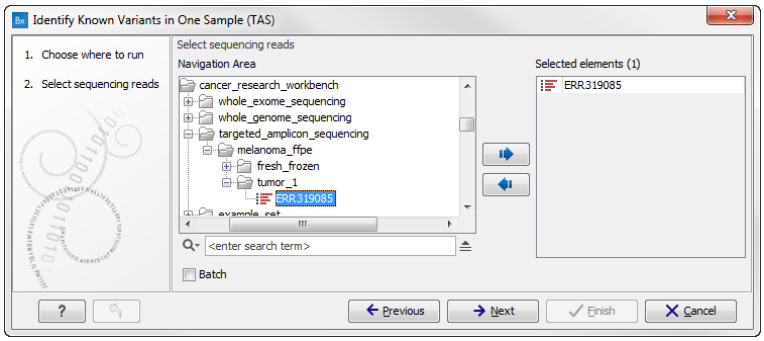


Figure 17.8: *Select the sequencing reads from the sample you would like to test for your known variants.*

   If several samples from different folders should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" and spcifying the folders that hold the data you wish to analyse.

   Click on the button labeled **Next**.

3. Specify the target region for the Indels and Structural Variants tool (figure 17.9). This step is optional and will speed the completion time of the workflow by running the tool only on the selected target regions. If you do not have a targeted region file to provide, simply click **Next**.

4. Specify the parameters for the QC for Target Sequencing tool (figure 17.10).

   When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. This step is not optional, and you need to specify the targeted regions file adapted to the sequencing technology you used. Choose to use the default settings or to adjust the parameters.

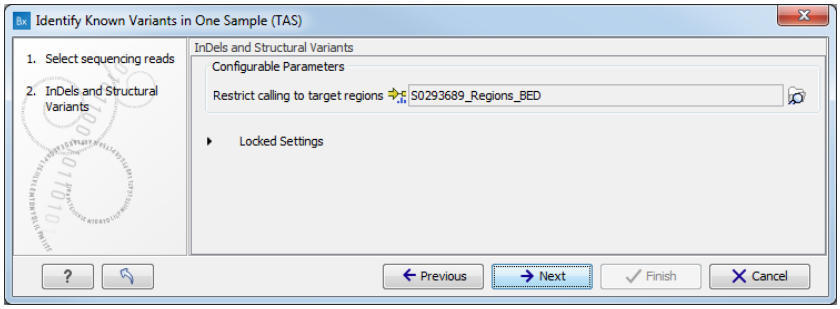


Figure 17.9: *Specify the targeted region file for the Indels and Structural Variants tool.*

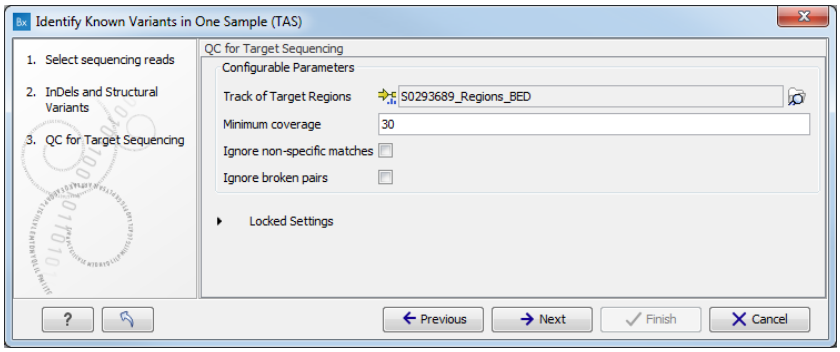


Figure 17.10: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

For more information about the tool, see section 20.1.

5. Click on the button labeled **Next** and specify the track with the known variants that should be identified in your sample (figure 17.11).

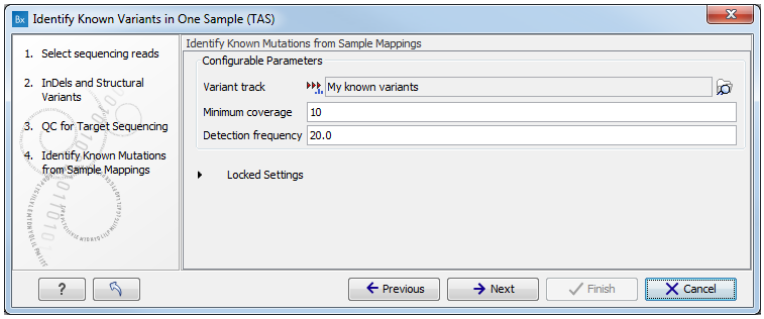


Figure 17.11: *Specify the track with the known variants that should be identified.*

The parameters that can be set are:

- **Minimum coverage** The minimum number of reads that covers the position of the variant, which is required to set "Sufficient Coverage" to YES.

- **Detection frequency** The minimum allele frequency that is required to annotate a variant as being present in the sample. The same threshold will also be used to determine if a variant is homozygous or heterozygous. In case the most frequent alternative allele at the position of the considered variant has a frequency of less than this value, the zygosity of the considered variant will be reported as being homozygous.

The parameter "Detection Frequency" will be used in the calculation twice. First, it will report in the result if a variant has been detected (observed frequency > specified frequency) or not (observed frequency <= specified frequency). Moreover, it will determine if a variant should be labeled as heterozygous (frequency of another allele identified at a position of a variant in the alignment > specified frequency) or homozygous (frequency of all other alleles identified at a position of a variant in the alignment < specified frequency).

Click on the button labeled **Next**.

6. In the last wizard step (figure 17.12)you can check the selected settings by clicking on the button labeled **Preview All Parameters**.

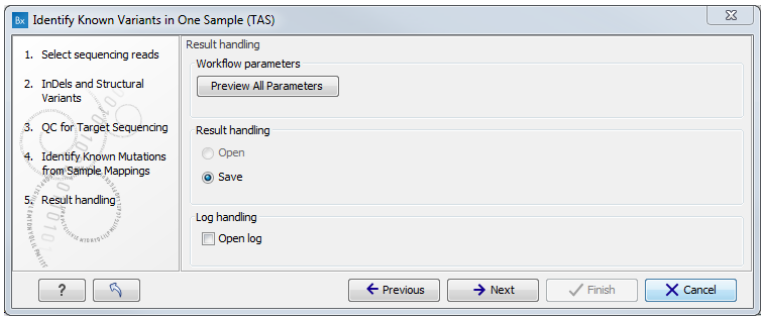


Figure 17.12: *Check the settings and save your results.*

At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination.

7. Click on the button labeled **OK** to go back to the previous dialog box and choose to **Save** your results.

Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Known Variants in One Sample (TAS)

The **Identify Known Variants in One Sample (TAS)** tool produces five different output types:

1. **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).

2. **Target Regions Coverage** ( ) A track showing the targeted regions. The table view provides information about the targeted regions such as target region length, coverage, regions without coverage, and GC content.

3. **Target Regions Coverage Report** ( ) The report consists of a number of tables and graphs that in different ways show e.g. the number, length, and coverage of the target regions and provides information about the read count per GC%.

4. **Variants Detected in Detail** ( ) Annotation track showing the known variants. Like the "Overview Variants Detected" table, this table provides information about the known variants. Four columns starting with the sample name and followed by "Read Mapping coverage", "Read Mapping detection", "Read Mapping frequency", and "Read Mapping zygosity" provides the overview of whether or not the known variants have been detected in the sequencing reads, as well as detailed information about the Most Frequent Alternative Allele (labeled MFAA).

5. **Genome Browser View Identify Known Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, target regions coverage, the mapped reads, the overview of the detected variants, and the variants detected in detail.

It is a good idea to start looking at the Target Regions Coverage Report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Please also check that at least 90% of the reads are mapped to the human reference sequence. In case of a targeted experiment, we also recommend that you check that the majority of the reads are mapping to the targeted region.

When you have inspected the target regions coverage report you can open the Genome Browser View Identify Known Variants file (see 17.13).

The Genome Browser View includes an overview track of the known variants and a detailed result track presented in the context of the human reference sequence, genes, transcripts, coding regions, targeted regions, and mapped sequencing reads.

Finally, a track with conservation scores has been added to be able to see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant.

By double clicking on one of the annotated variant tracks in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see 17.14).

Note We do not recommend that any of the produced files are deleted individually as some of them are linked to other outputs. Please always delete all of them at the same time.

17.2 Somatic Cancer (TAS)

17.2.1 Filter Somatic Variants (TAS)

If you are analyzing a list of variants that have been detected in a tumor or blood sample where no control sample is available from the same patient, you can use the **Filter Somatic Variants (TAS)** ready-to-use workflow to identify potential somatic variants. The purpose of this ready-to-use workflow is to use publicly available (or your own) databases, with common variants in a population, to extract potential somatic variants whenever no control/normal sample from the same patient is available.

The **Filter Somatic Variants (TAS)** ready-to-use workflow accepts variant tracks ( ) (e.g. the output from the Identify Variants ready-to-use workflow) as input. Variants that are identical to the

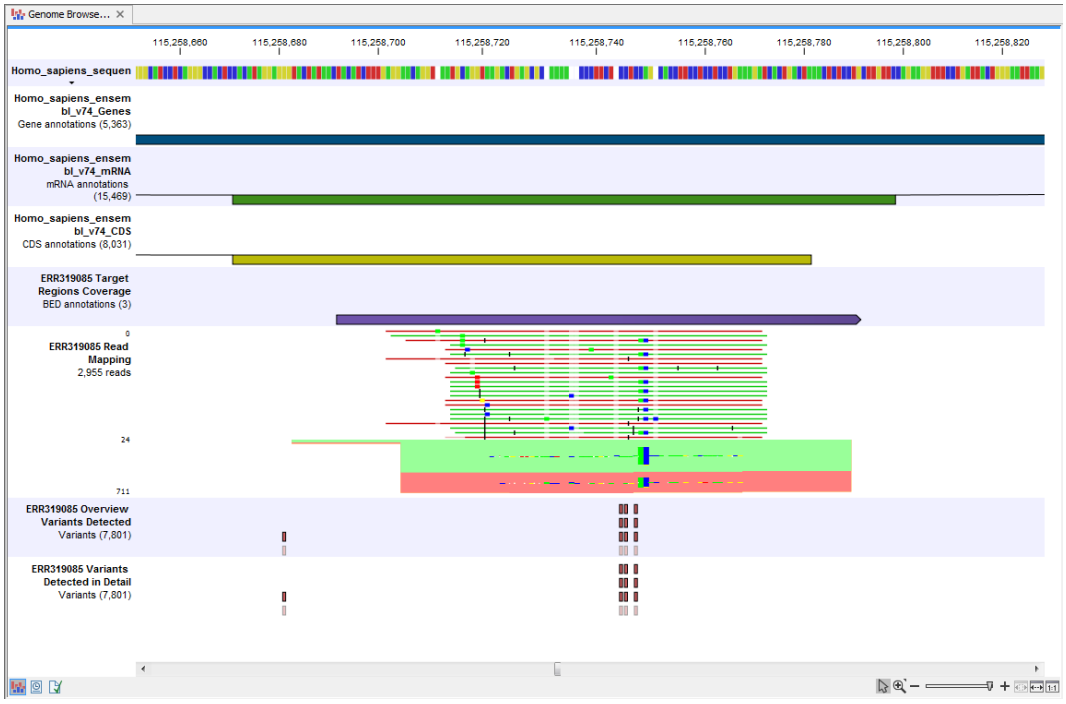


Figure 17.13: *Genome Browser View that allows inspection of the identified variants in the context of the human genome and external databases.*

human reference sequence are first filtered away, then variants outside the targeted region are removed, and lastly, variants found in the Common dbSNP, 1000 Genomes Project, and HapMap databases are deleted. Variants in those databases are assumed to not contain relevant somatic variants.

Please note that this tool will likely also remove inherited cancer variants that are present at a low percentage in a population.

Next, the remaining somatic variants are annotated with gene names, amino acid changes, conservation scores and information from ClinVar (known variants with medical impact) and dbSNP (all known variants).

How to run the Filter Somatic Variants (TAS) workflow

To run the **Filter Somatic Variants (TAS)** tool, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing** ( ) | **Somatic Cancer** ( ) | **Filter Somatic Variants** ( )

1. Double-click on the **Filter Somatic Variants (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Next, you will be asked to select the variant track you would like to use for filtering somatic variants. The panel in the left side of the wizard shows the kind of input that should be provided (figure 17.15). Select by double-clicking on the reads file name or clicking once on the file and then clicking on the arrow pointing to the right side in the middle of the wizard.

   Click on the button labeled **Next**.

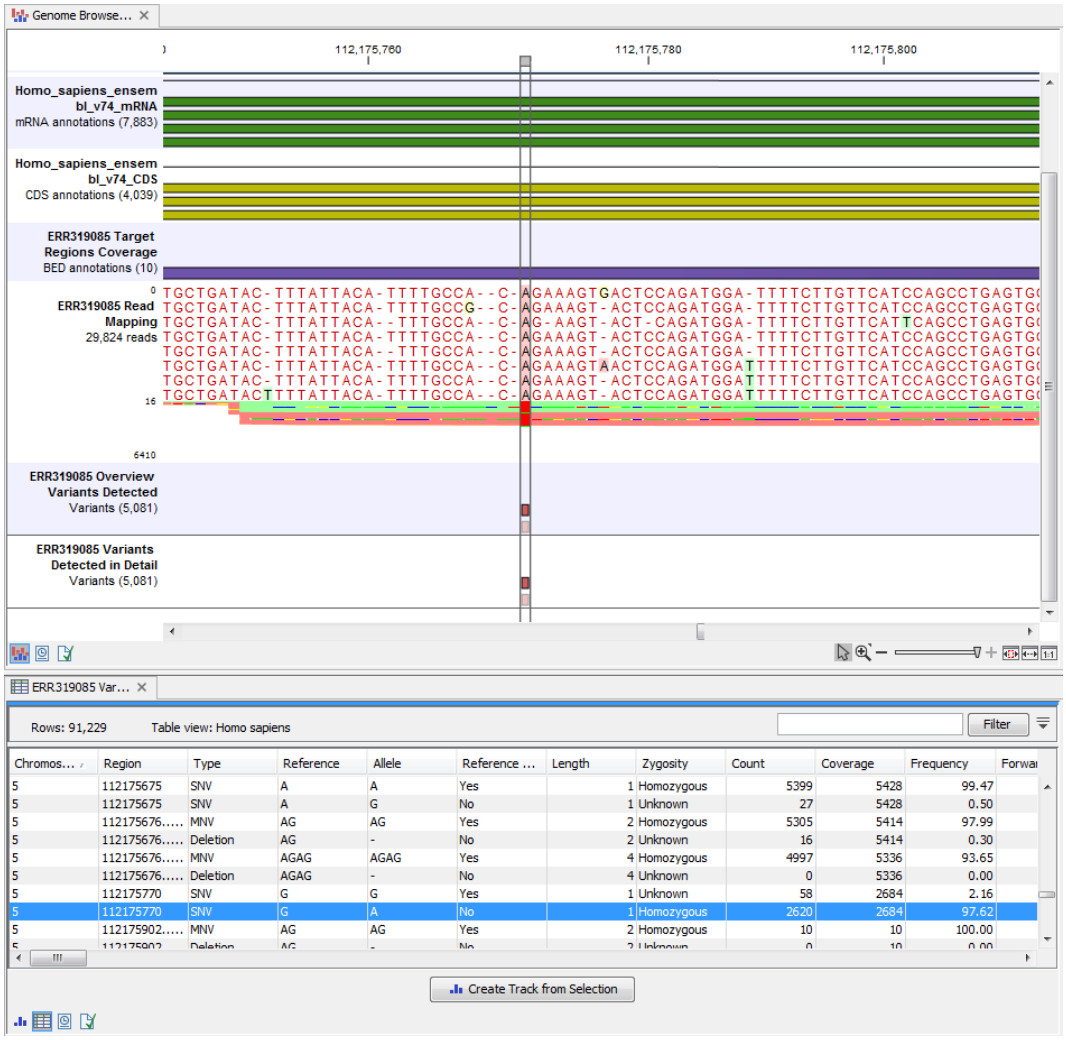


Figure 17.14: *Genome Browser View with an open overview variant track with information about if the variant has been detected or not, the identified zygosity, if the coverage was sufficient at this position and the observed allele frequency.*

3. In the next step you will be asked to specify which of the 1000 Genomes populations that should be used for annotation (figure 17.16).

   Click on the button labeled **Next**.

4. The next wizard step will once again allow you to specify the 1000 Genomes population that should be used, this time for filtering out variants found in the 1000 Genomes project (figure 17.17).

   Click on the button labeled **Next**.

5. The next wizard step (figure 17.18) concerns removal of variants found in the HapMap database. Select the population you would like to use from the drop-down list. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

6. Click on the button labeled **Next** to go to the last wizard step (shown in figure 17.19).

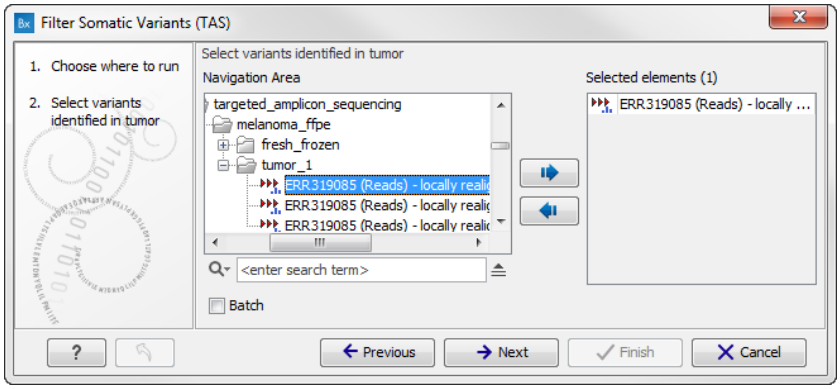


Figure 17.15: *Select the variant track from which you would like to filter somatic variants.*

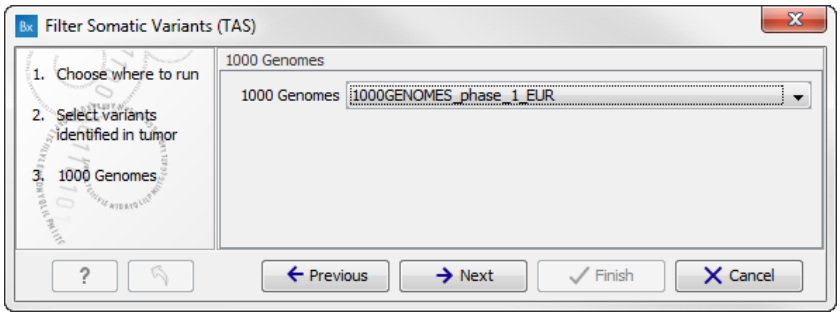


Figure 17.16: *Specify which 1000 Genomes population to use for annotation.*

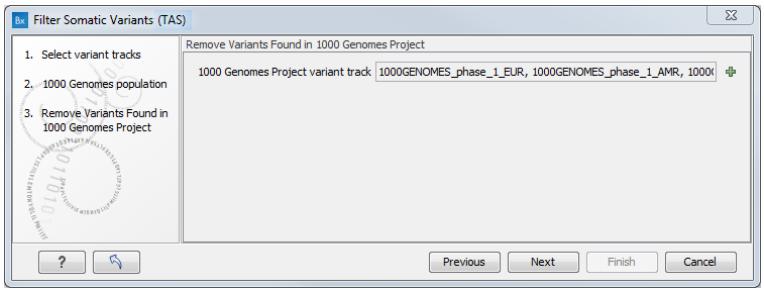


Figure 17.17: *Specify which 1000 Genomes population to use for filtering out known variants.*

Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Filter Somatic Variants (TAS) workflow

Two types of output are generated:

1. **Amino Acids Changes** Track that shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.

2. **Somatic Candidate Variants** Track that holds the variant data. This track is also included in the Genome Browser View. If you hold down the Ctrl key (Cmd on Mac) while clicking on

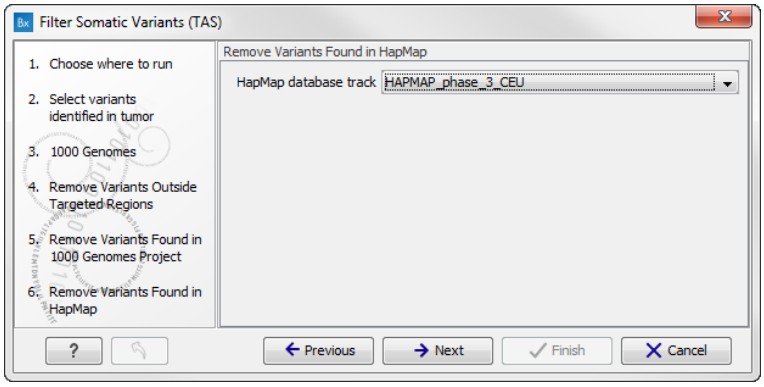


Figure 17.18: *Specify which HapMap population to use for filtering out known variants.*

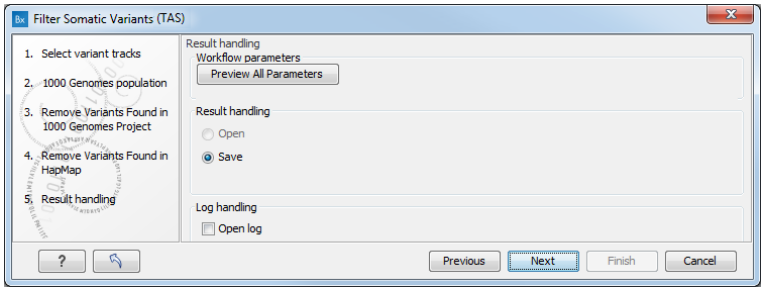


Figure 17.19: *Check the selected parametes by pressing "Preview All Parameters".*

the table icon in the lower left side of the **View Area**, you can open the table view in split view. The table and the variant track are linked together, and when you click on a row in the table, the track view will automatically bring this position into focus.

3. **Genome Browser View Filter Somatic Variants** A collection of tracks presented together. Shows the somatic candidate variants together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar, 1000 Genomes, and the PhastCons conservation scores (see figure 17.20).

To see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant, a track with conservation scores is added as well. Mapped sequencing reads as well as other tracks can be easily added to this Genome Browser View. By double clicking on the annotated variant track in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see figure 17.21).

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar databases, can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons

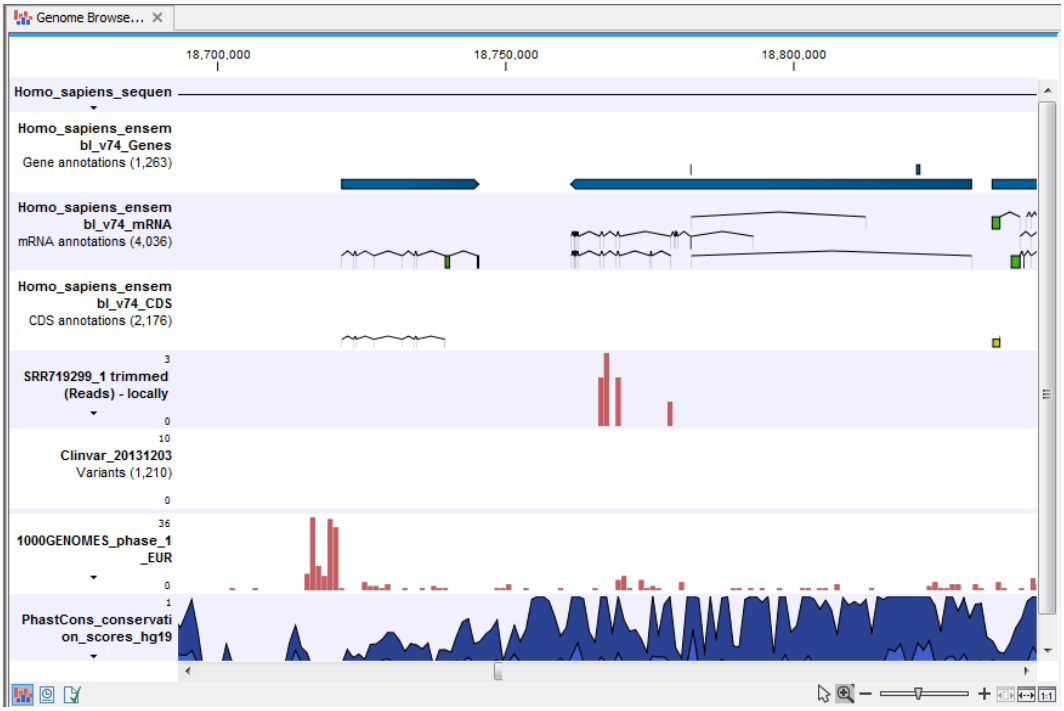


Figure 17.20: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** ( ) | **Create Filter Criteria** ( )

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

17.2.2 Identify Somatic Variants from Tumor Normal Pair (TAS)

The **Identify Somatic Variants from Tumor Normal Pair (TAS)** ready-to-use workflow can be used to identify potential somatic variants in a tumor sample when you also have a normal/control sample from the same patient.

When running the **Identify Somatic Variants from Tumor Normal Pair (TAS)** the reads are mapped and the variants identified. An internal workflow removes germline variants that are found in the mapped reads of the normal/control sample and variants outside the target region are removed as they are likely to be false positives due to non-specific mapping of sequencing reads. Next,

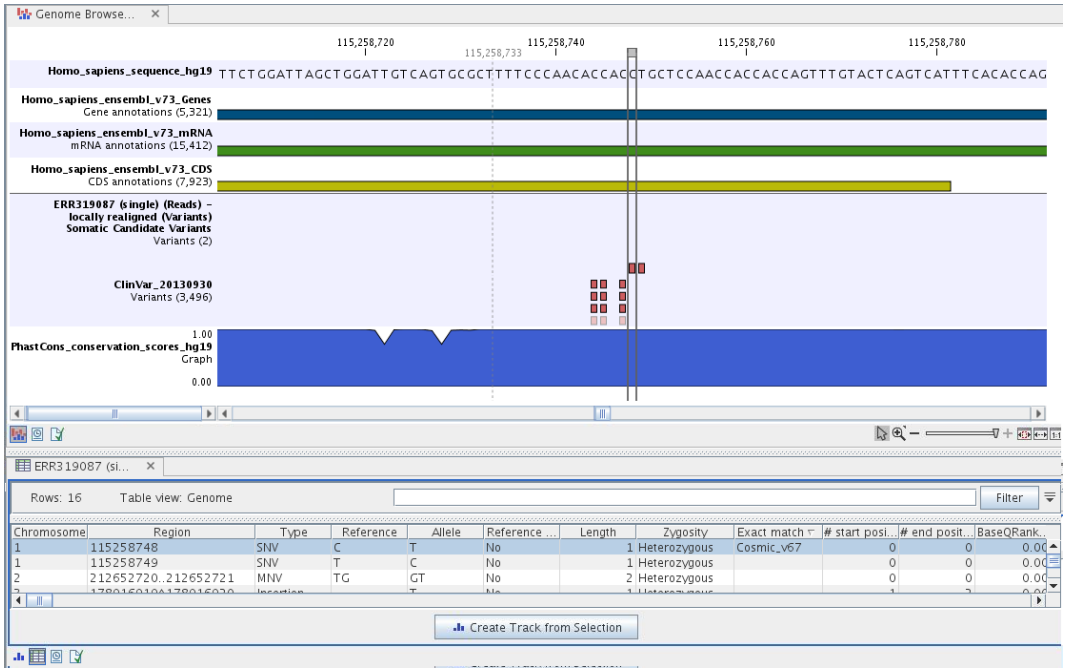


Figure 17.21: *The Genome Browser View showing the annotated somatic variants together with a range of other tracks.*

remaining variants are annotated with gene names, amino acid changes, conservation scores and information from relevant databases like ClinVar (variants with clinically relevant association). Finally, information from dbSNP is added to see which of the detected variants have been observed before and which are completely new.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit is available from the vendor of the enrichment kit and sequencing machine. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get the file in either .bed or .gff format.

To import the file:

Go to the toolbar | Import () | Tracks ()

How to run the Identify Somatic Variants from Tumor Normal Pair (TAS) workflow

1. Go to the toolbox and double-click on the **Identify Somatic Variants from Tumor Normal Pair (TAS)** ready-to-use workflow. This will open the wizard shown in figure 17.22 where you can select the tumor sample reads.

   When you have selected the tumor sample reads click on the button labeled **Next**.

2. In the next wizard step (figure 17.23), please specify the normal sample reads.

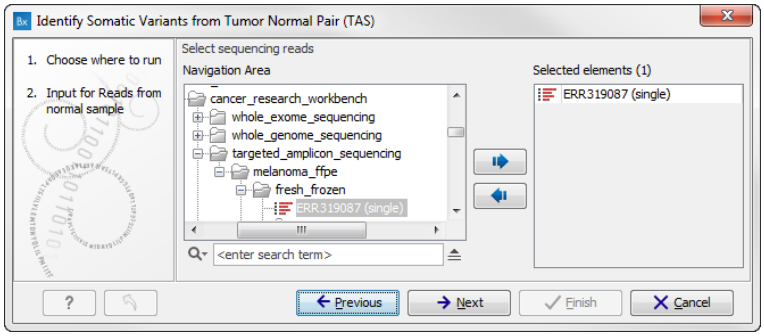


Figure 17.22: *Select the tumor sample reads.*

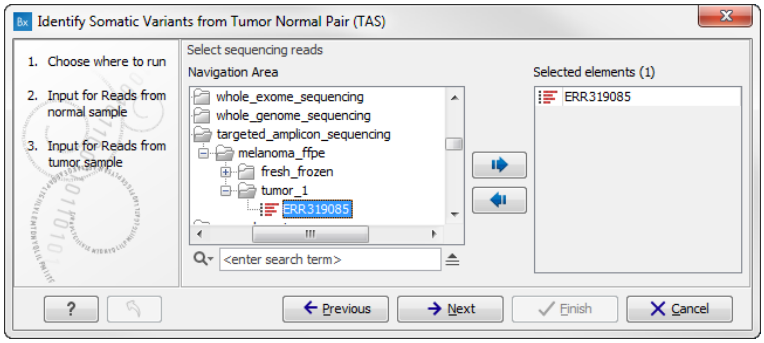


Figure 17.23: *Select the normal sample reads.*

3. The following 2 steps allow you to restrict the calling of indels and structural variants to the targeted regions, both for tumor and normal reads (figure 17.24).

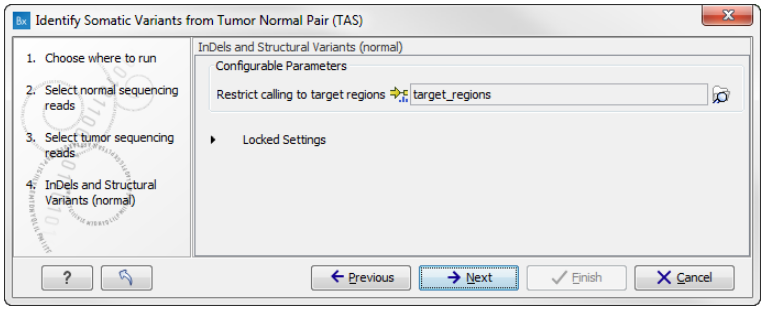


Figure 17.24: *Specify the target regions track.*

4. Set the parameters for the Low Frequency Variant Detection step (figure 17.25) and click **Next**.

5. In the following 2 wizard steps, you can select your target regions track to be used for reporting the performance of the targeted re-sequencing experiment for the tumor and normal samples successively (figure 17.26). The targeted region track should be the same as the track you selected in the previous wizard steps. Variants found outside the targeted regions will not be included in the output that is generated with the ready-to-use workflow.

6. Next, adjust the settings for removal of germline variants step (figure 17.27). Click on the button labeled **Next**.

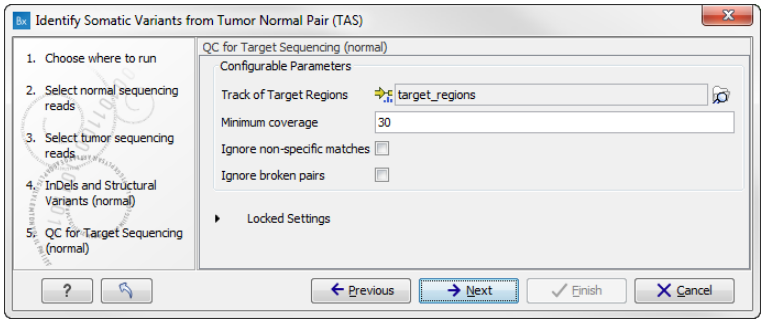


Figure 17.25: *Specify the settings for the variant detection.*

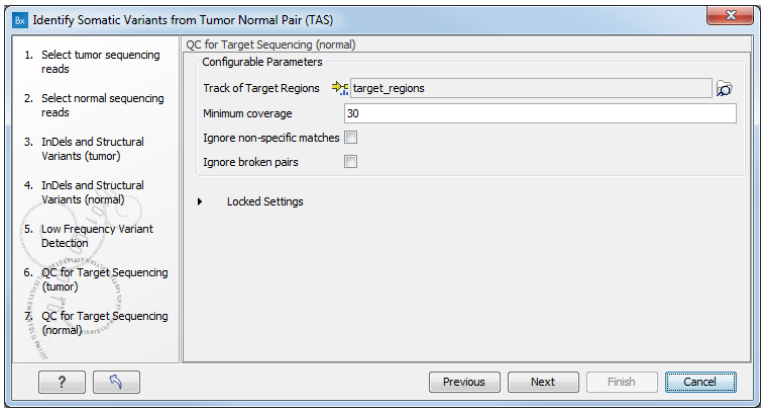


Figure 17.26: *Select your target region track.*

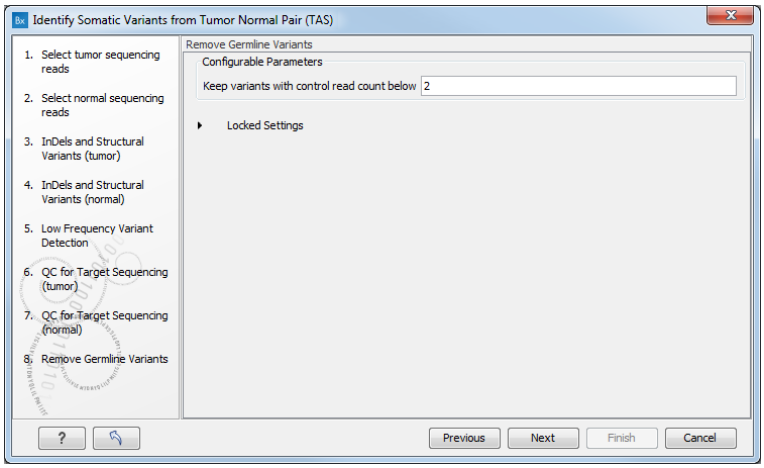


Figure 17.27: *Select your target region track.*

7. In the next wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters** (figure 17.28).

   In the **Preview All Parameters** wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When

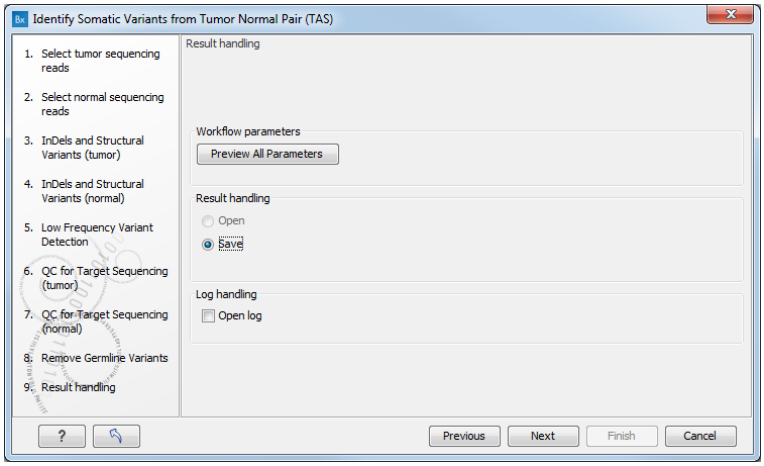


Figure 17.28: *Check the parameters and save the results.*

selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

8. Click on the button labeled **OK** to go back to the previous wizard step and choose **Save**.

Output from the Identify Somatic Variants from Tumor Normal Pair (TAS) workflow

Nine different outputs are generated:

- **Read Mapping Normal** ( ) The mapped sequencing reads for the normal sample. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).
- **Read Mapping Tumor** ( ) The mapped sequencing reads for the tumor sample.
- **Target Region Coverage Report Normal** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from the normal sample.
- **Target Region Coverage Tumor** ( ) A track showing the targeted regions. The table view provides information about the targeted regions such as target region length, coverage, regions without coverage, and GC content.
- **Target Region Coverage Report Tumor** ( ) The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from the tumor sample.
- **Amino Acids Changes** Track that shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Variants** ( ) A variant track holding the identified variants that are found in the targeted regions. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

- **Annotated Somatic Variants** ( ) A variant track holding the identified and annotated somatic variants. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.

- **Genome Browser View Tumor Normal Comparison** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads for both normal and tumor, the annotated somatic variants, information from the ClinVar database, and finally a track showing the conservation score (see figure 17.29).

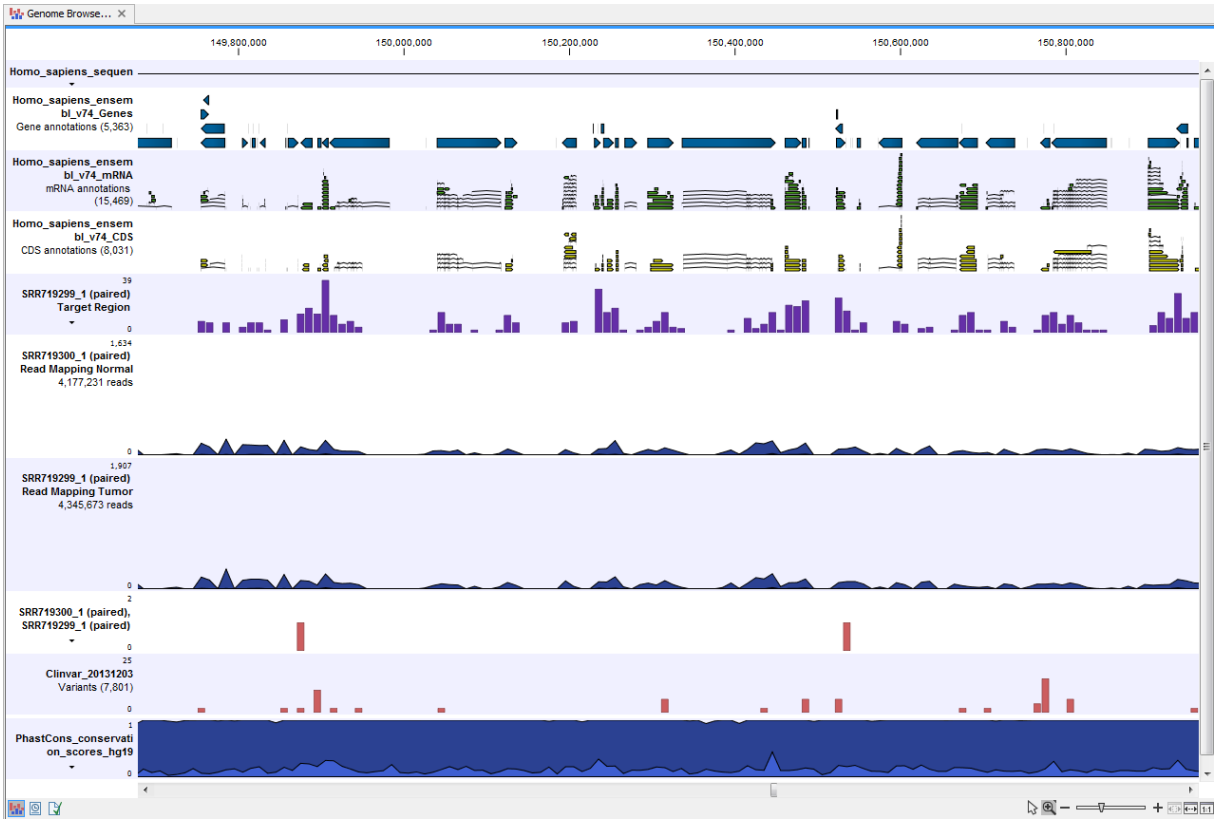


Figure 17.29: *The Genome Browser View presents all the different data tracks together and makes it easy to compare different tracks.*

17.2.3 Identify Variants (TAS)

The **Identify Variants (TAS)** tool takes sequencing reads as input and returns identified variants as part of a Genome Browser View.

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection, which is run afterwards. At the end, variants with an average base quality smaller than 20 are filtered away.

In addition, a targeted region report is created to inspect the overall coverage and mapping specificity in the targeted regions.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit will be provided by the vendor. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get it in either .bed or .gff format.

Please use the Tracks import as part of the Import tool in the toolbar to import your file into the *Biomedical Genomics Workbench*.

How to run the Identify Variants (TAS) workflow

To run the **Identify Variants (TAS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Variants (TAS** ()

1. Select the sequencing reads from the sample that should be analyzed (figure 17.30).

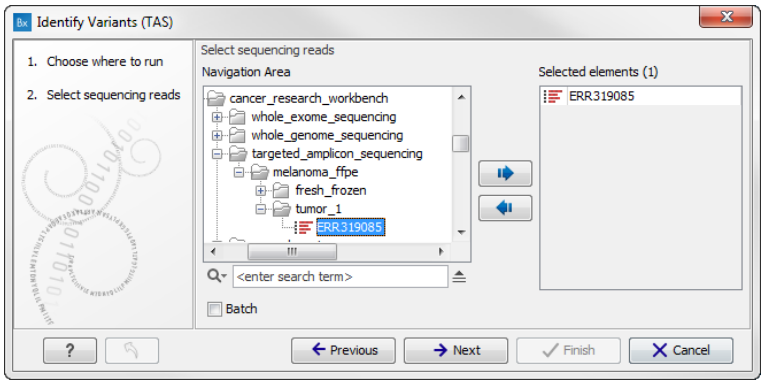


Figure 17.30: *Please select all sequencing reads from the sample to be analyzed.*

Select all sequencing reads from your sample. If several samples should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" (tick "Batch" at the bottom of the wizard as shown in figure 17.38) and select the **folder** that holds the data you wish to analyze. If you have your sequencing data in separate folders, you should choose to run the analysis in batch mode.

When you have selected the sample(s) you wish to prepare, click on the button labeled **Next**.

2. In this wizard you can restrict calling of indels and structural variants to the targeted regions by specifying the track with the targeted regions from the experiment (figure 17.31).

3. In the next wizard step (figure 17.32) you have to specify the track with the targeted regions from the experiment. You can also specify the minimum read coverage, which should be present in the targeted regions.

4. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.33). In this wizard you can specify the parameter for detecting variants.

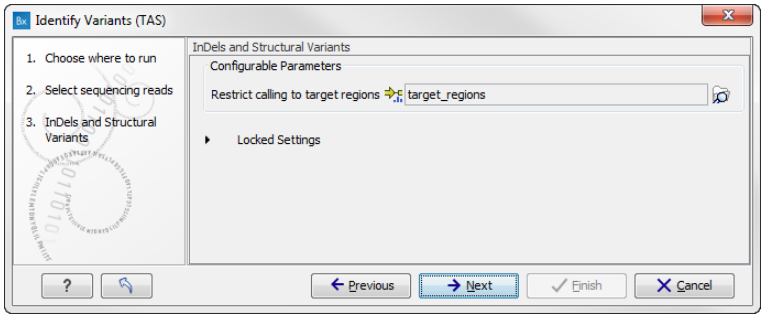


Figure 17.31: *Select the track with the targeted regions from your experiment.*

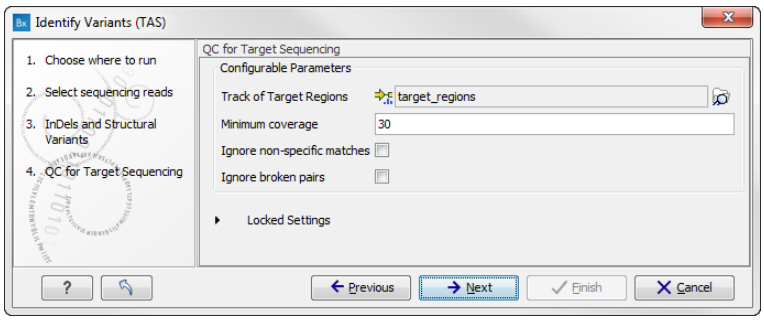


Figure 17.32: *Select the track with the targeted regions from your experiment.*

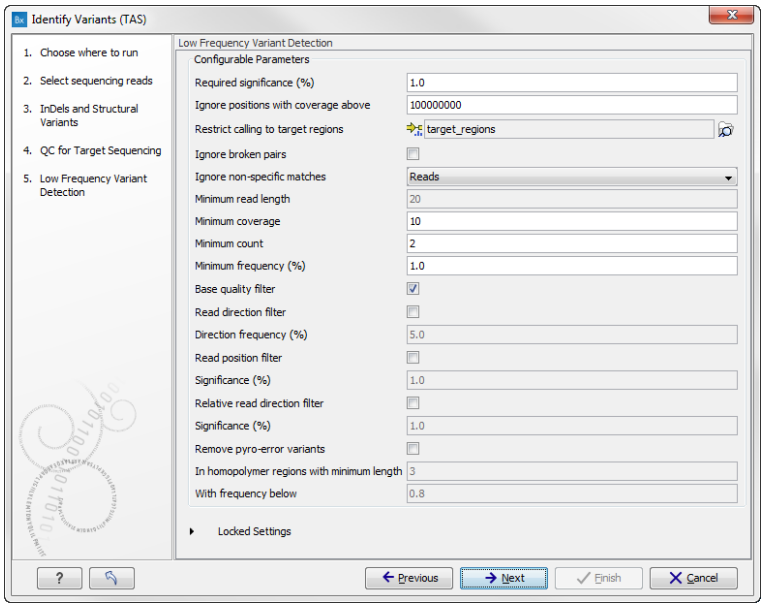


Figure 17.33: *Please specify the parameters for variant detection.*

5. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.34).

6. Click on the button labeled **Next** to go to the last wizard step (figure 17.35).

   In this wizard you get the chance to check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard step you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows. At the bottom of this

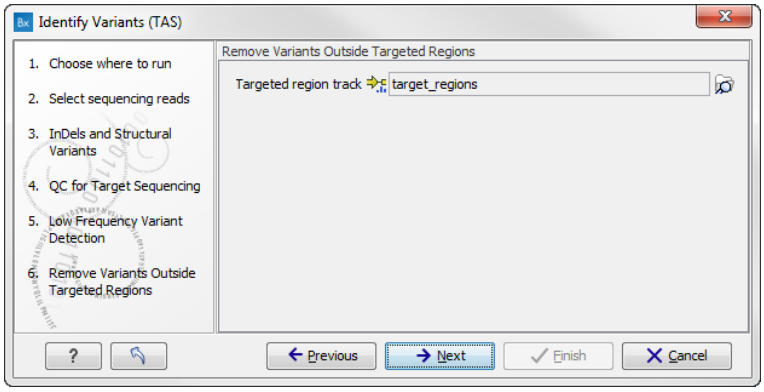


Figure 17.34: *Select the targeted region track. Variants found outside the targeted region will be removed.*

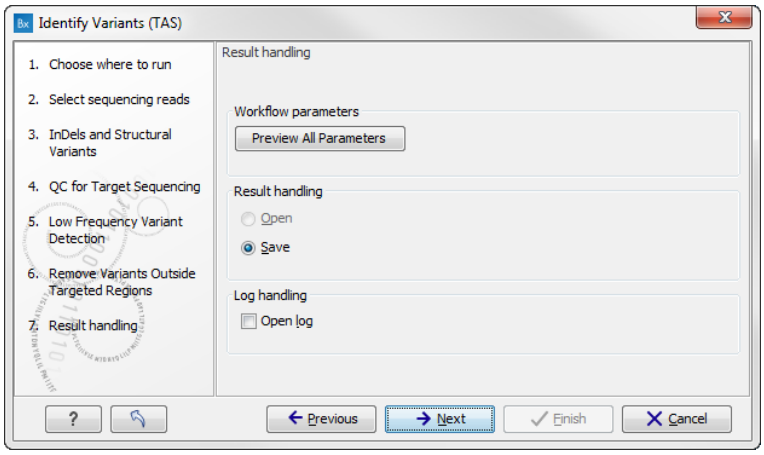


Figure 17.35: *Choose to save the results. In this wizard step you get the chance to preview the settings used in the ready-to-use workflow.*

wizard there are two buttons regarding export functions; one button allows specification of the export format, and the other button (the one labeled "Export Parameters") allows specification of the export destination. When selecting an export location, you will export the analysis parameter settings that were specified for this specific experiment.

7. Click on the button labeled **OK** to go back to the previous wizard step and choose **Save**.

   Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify Variants (TAS) workflow

The **Identify Variants (TAS)** tool produces six different types of output:

- **Read Mapping** ( ) The mapped sequencing reads. The reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously (see section 22.2.2).
- **Target Regions Coverage** ( ) The target regions coverage track shows the coverage of the targeted regions. Detailed information about coverage and read count can be found in the

table format, which can be opened by pressing the table icon found in the lower left corner of the **View Area**.

- **Target Regions Coverage Report** ( ) The report consists of a number of tables and graphs that in different ways provide information about the targeted regions.
- **Identified Variants** ( ) A variant track holding the identified variants. The variants can be shown in track format or in table format. When holding the mouse over the detected variants in the Genome Browser view a tooltip appears with information about the individual variants. You will have to zoom in on the variants to be able to see the detailed tooltip.
- **Genome Browser View Identify Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, the mapped reads, the identified variants, and the structural variants (see figure 17.36).

It is important that you do not delete any of the produced files individually as some of the outputs are linked to other outputs. If you would like to delete the outputs, please always delete all of them at the same time.

Please have first a look at the mapping report to see if the coverage is sufficient in regions of interest (e.g. > 30). Furthermore, please check that at least 90% of reads are mapped to the human reference sequence. In case of a targeted experiment, please also check that the majority of reads are mapping to the targeted region.

Afterwards please open the Genome Browser View file (see 17.36).

The Genome Browser View includes the track of identified variants in context to the human reference sequence, genes, transcripts, coding regions, targeted regions and mapped sequencing reads.

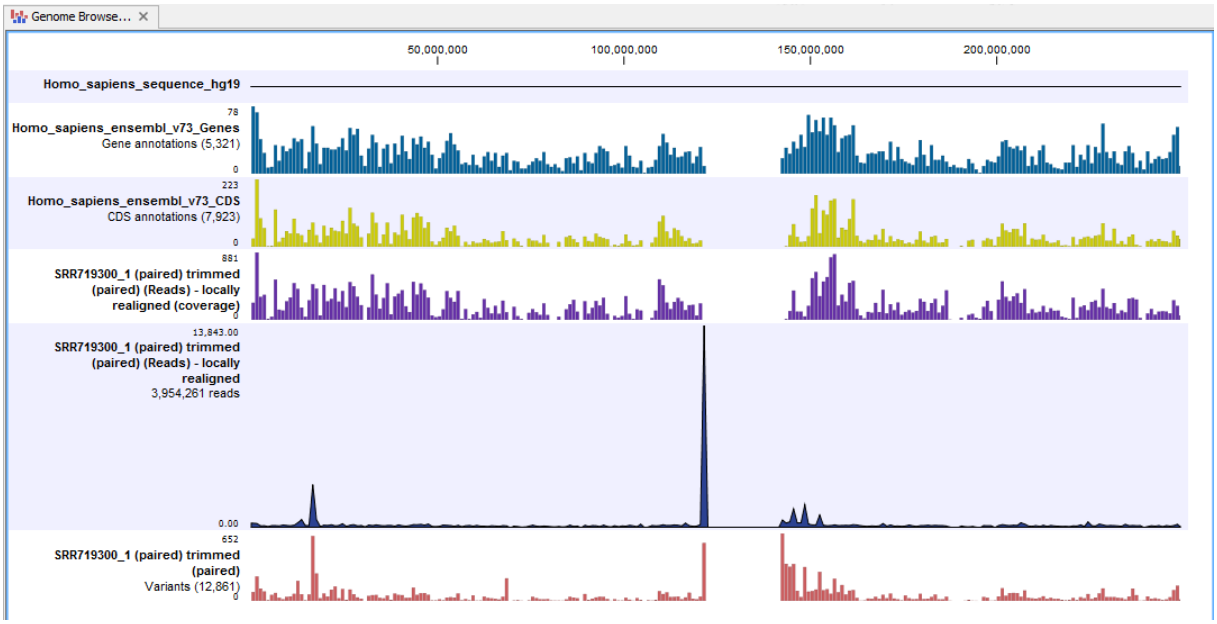


Figure 17.36: *The Genome Browser View allows you to inspect the identified variants in the context of the human genome.*

By double clicking on the variant track in the Genome Browser View, a table will be shown which includes information about all identified variants (see 17.37).

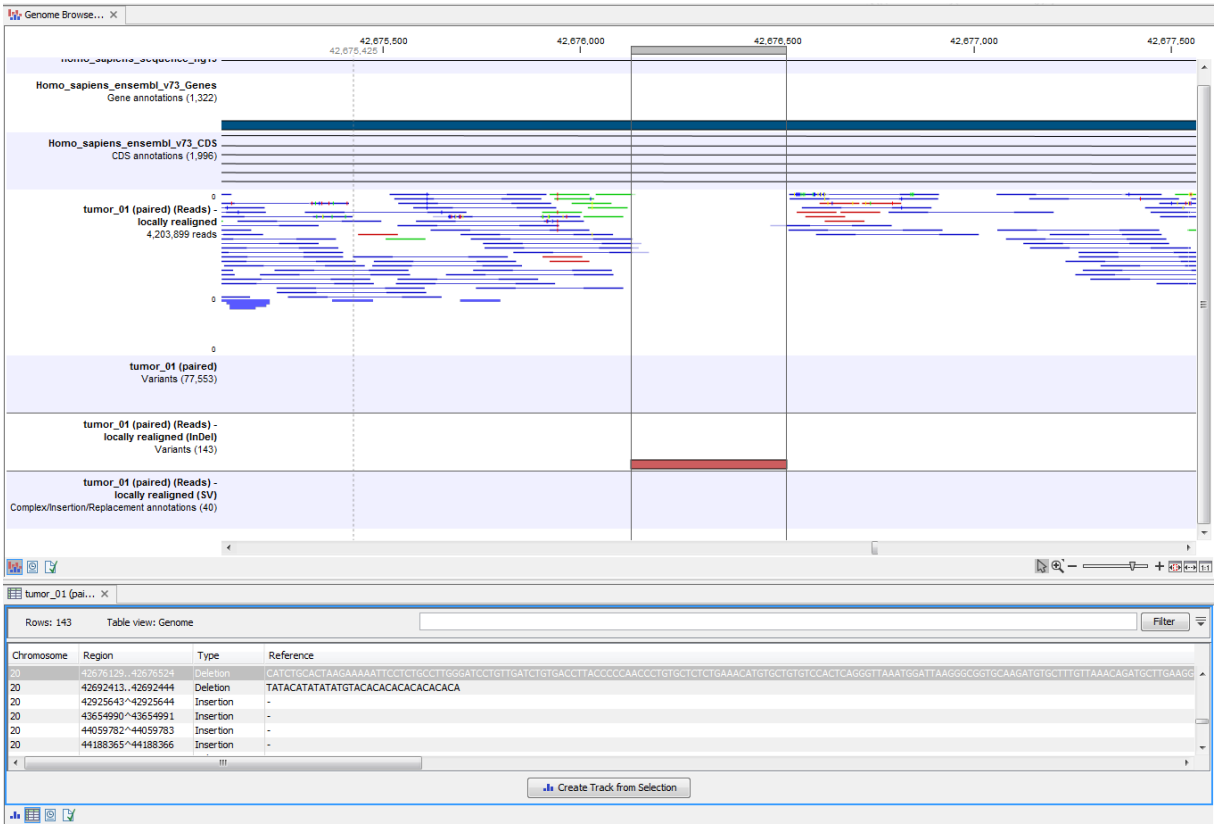


Figure 17.37: *Genome Browser View with an open track table to inspect identified variants more closely in the context of the human genome.*

In case you like to change the reference sequence used for mapping as well as the human genes, please use the "Data Management".

17.2.4 Identify and Annotate Variants (TAS)

The **Identify and Annotate Variants (TAS)** tool should be used to identify and annotate variants in one sample. The tool consists of a workflow that is a combination of the **Identify Variants** and the **Annotate Variants** workflows.

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection, which is run afterwards. After the variants have been detected, they are annotated with gene names, amino acid changes, conservation scores, information from relevant variants present in the ClinVar database, and information from common variants present in the common dbSNP, HapMap, and 1000 Genomes database. Furthermore, a detailed mapping report or a targeted region report (whole exome and targeted amplicon analysis) is created to inspect the overall coverage and mapping specificity.

Import your targeted regions

A file with the genomic regions targeted by the amplicon or hybridization kit is available from the vendor of the enrichment kit and sequencing machine. To obtain this file you will have to get in contact with the vendor and ask them to send this target regions file to you. You will get the file in either .bed or .gff format.

To import the file:

Go to the toolbar | Import ( ) | Tracks ( )

How to run the Identify and Annotate Variants (TAS) workflow

To run the **Identify and Annotate Variants (TAS)** workflow, go to:

Toolbox | Ready-to-Use Workflows | Targeted Amplicon Sequencing Sequencing ( ) | Somatic Cancer ( ) | Identify and annotate Variants (TAS) ( )

1. Double-click on the **Identify and Annotate Variants (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis. Click on the button labeled Next.

2. This will open the wizard shown in figure 17.38 where you can select the sequencing reads from the sample that should be analyzed.

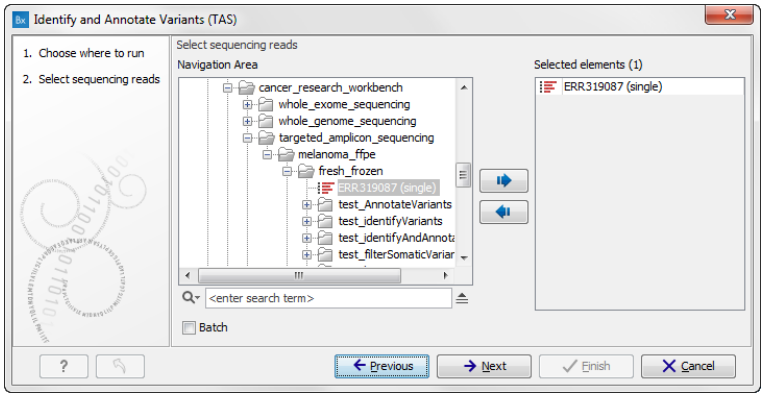


Figure 17.38: *Please select all sequencing reads from the sample to be analyzed.*

If several samples should be analyzed, the tool has to be run in batch mode. This is done by selecting "Batch" (tick "Batch" at the bottom of the wizard as shown in figure 17.38) and select the **folder** that holds the data you wish to analyse. If you have your sequencing data in separate folders, you should choose to run the analysis in batch mode.

When you have selected the sample(s) you wish to prepare, click on the button labeled **Next**.

3. In the next wizard step (figure 17.39) you can select the population from the 1000 Genomes project that you would like to use for annotation.

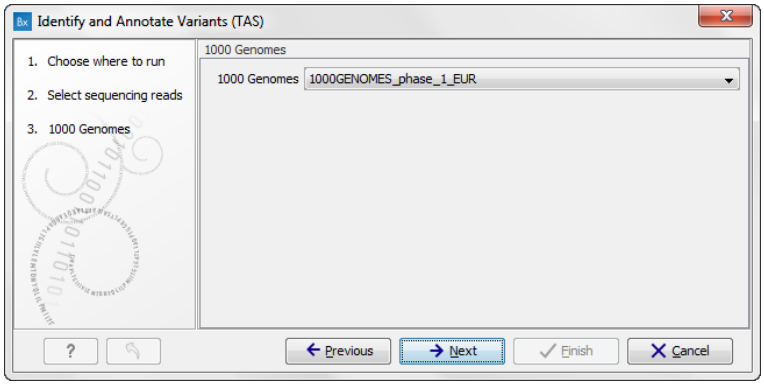


Figure 17.39: *Select the population from the 1000 Genomes project that you would like to use for annotation.*

4. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.40). In this dialog you can specify the target regions track. The variants found outside the targeted region will be removed at this step in the workflow.

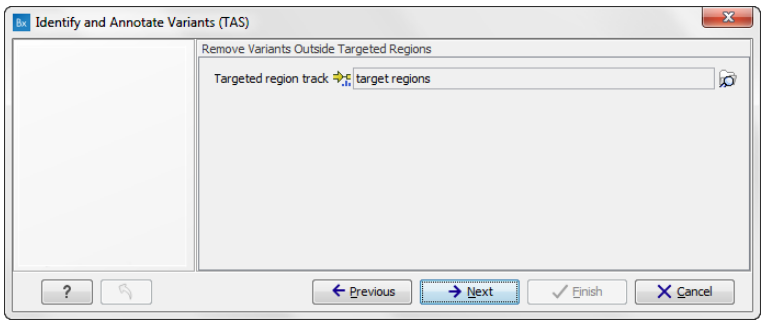


Figure 17.40: *In this wizard step you can specify the target regions track. Variants found outside these regions will be removed.*

5. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.41). In this dialog, you have to specify the parameters for the variant detection. For a description of the different parameters that can be adjusted, see section 22.14. If you click on "Locked Settings", you will be able to see all parameters used for variant detection in the ready-to-use workflow.

6. In the next wizard (figure 17.42) you can select the target region track and specify the minimum read coverage that should be present in the targeted regions.

7. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.43). Once again, select the relevant population from the 1000 Genomes project. This will add information from the 1000 Genomes project to your variants.

8. Click on the button labeled **Next**, which will take you to the next wizard step (figure 17.44). At this step you can select a population from the HapMap database. This will add information from the Hapmap database to your variants.

9. In this wizard step (figure 17.45) you get the chance to check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters**

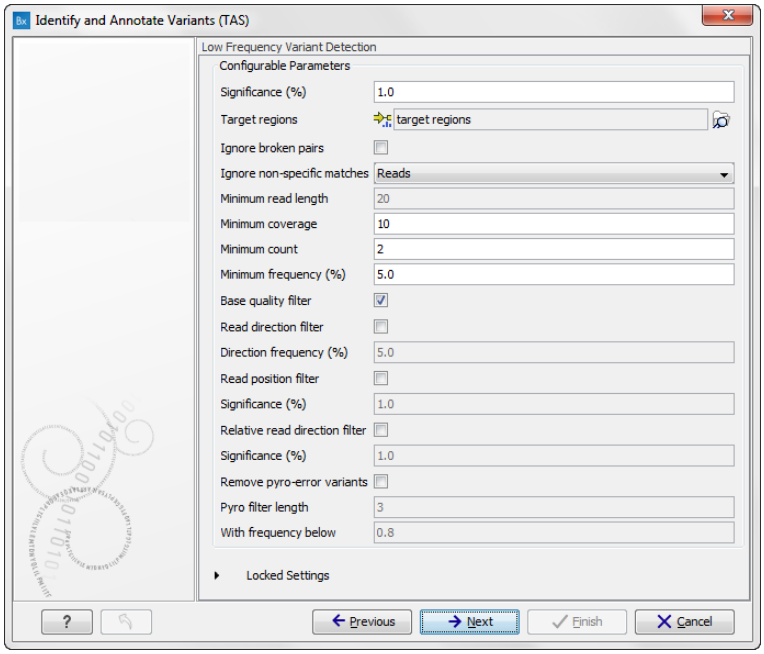


Figure 17.41: *Specify the parameters for variant calling.*

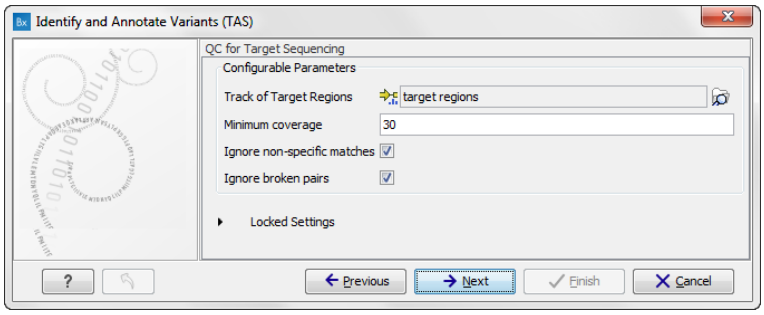


Figure 17.42: *Select the track with targeted regions from your experiment.*

wizard you can only check the settings, and if you wish to make changes you have to use the **Previous** button from the wizard to edit parameters in the relevant windows.

10. Choose to **Save** your results and press **Finish**.

    Note! If you choose to open the results, the results will not be saved automatically. You can always save the results at a later point.

Output from the Identify and Annotate Variants (TAS) workflow

The **Identify and Annotate Variants (TAS)** tool produces several outputs.

Please do not delete any of the produced files alone as some of them are linked to other outputs. Please always delete all of them at the same time.

A good place to start is to take a look at the mapping report to see whether the coverage is sufficient in the regions of interest (e.g. > 30). Furthermore, please check that at least 90% of the reads are mapped to the human reference sequence. In case of a targeted experiment, please also check that the majority of the reads are mapping to the targeted region.

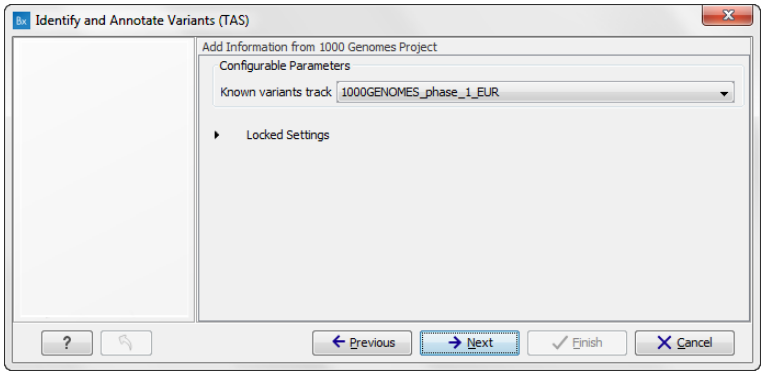


Figure 17.43: *Select the relevant population from the 1000 Genomes project. This will add information from the 1000 Genomes project to your variants.*

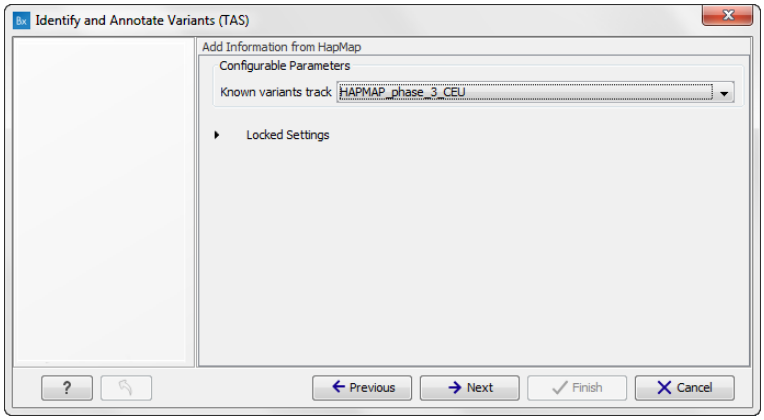


Figure 17.44: *Select a population from the HapMap database. This will add information from the Hapmap database to your variants.*

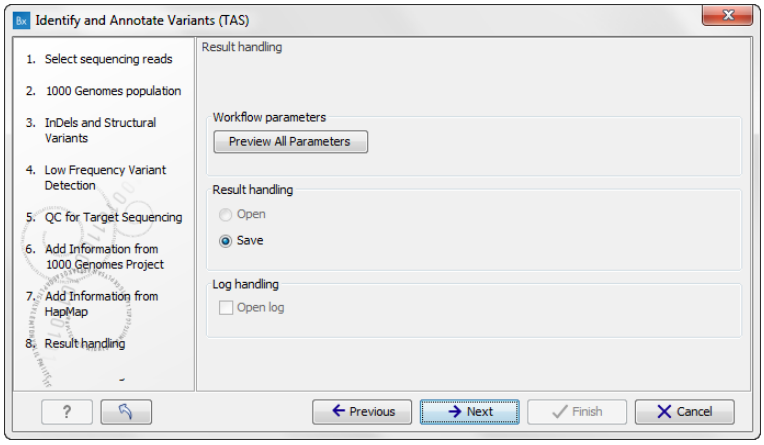


Figure 17.45: *Check the settings and save your results.*

Next, open the Genome Browser View file (see figure 17.46).

The Genome Browser View includes a track of the identified annotated variants in context to the human reference sequence, genes, transcripts, coding regions, targeted regions, mapped sequencing reads, relevant variants in the ClinVar database as well as common variants in

common dbSNP, HapMap, and 1000 Genomes databases.

Figure 17.46: *Genome Browser View to inspect identified variants in the context of the human genome and external databases.*

To see the level of nucleotide conservation (from a multiple alignment with many vertebrates) in the region around each variant, a track with conservation scores is added as well.

By double-clicking on the annotated variant track in the Genome Browser View, a table will be shown that includes all variants and the added information/annotations (see figure 17.47).

The added information will help you to identify candidate variants for further research. For example can common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) easily be seen.

Not identified variants in ClinVar, can for example be prioritized based on amino acid changes (do they cause any changes on the amino acid level?). A high conservation level on the position of the variant between many vertebrates or mammals can also be a hint that this region could have an important functional role and variants with a conservation score of more than 0.9 (PhastCons score) should be prioritized higher. A further filtering of the variants based on their annotations can be facilitated using the table filter on top of the table.

If you wish to always apply the same filter criteria, the "Create new Filter Criteria" tool should be used to specify this filter and the "Identify and Annotate" workflow should be extended by the "Identify Candidate Tool" (configured with the Filter Criterion). See the reference manual for more information on how preinstalled workflows can be edited.

Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

In case you like to change the databases as well as the used database version, please use the "Data Management".

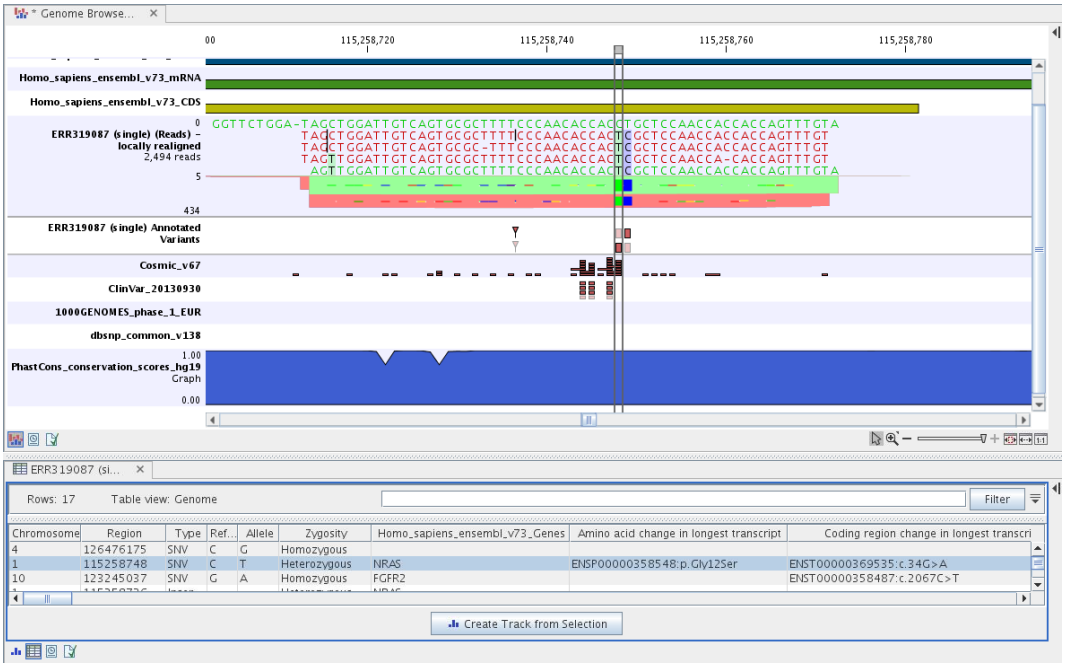


Figure 17.47: *Genome Browser View with an open track table to inspect identified somatic variants more closely in the context of the human genome and external databases.*

17.3 Hereditary Disease (TAS)

17.3.1 Filter Causal Variants (TAS-HD)

If you are analyzing a list of variants, you can use the **Filter Causal Variants (TAS-HD)** ready-to-use workflow to remove variants that are outside the target region, as well as common variants present in publicly available databases. The workflow will annotate the remaining variants with gene names, conservation scores, and information from relevant databases.

The **Filter Causal Variants (TAS-HD)** ready-to-use workflow accepts variants tracks files as input files.

How to run the Filter Causal Variants (TAS-HD) workflow

To run the **Filter Causal Variants (TAS-HD))** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Filter Candidate Variants (TAS - HD)** ()

1. Double-click on the **Filter Causal Variants (TAS-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the **variant track** you want to use for filtering causal variants (figure 17.48). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the variant track name or click once on the file and then click on the arrow pointing to the right side in the middle of the wizard.
3. Specify which of the **1000 Genomes populations** that should be used for **annotation** (figure 17.49).

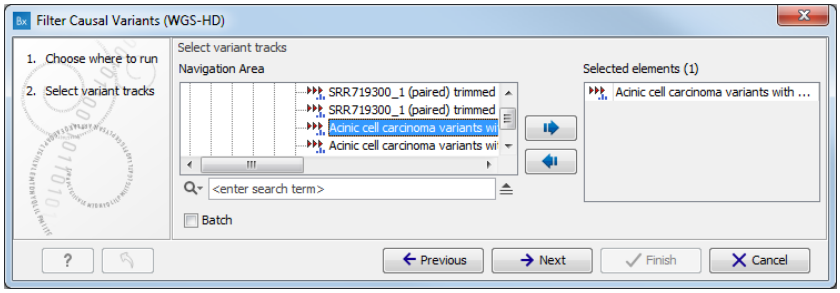


Figure 17.48: *Select the variant track from which you would like to filter somatic variants.*

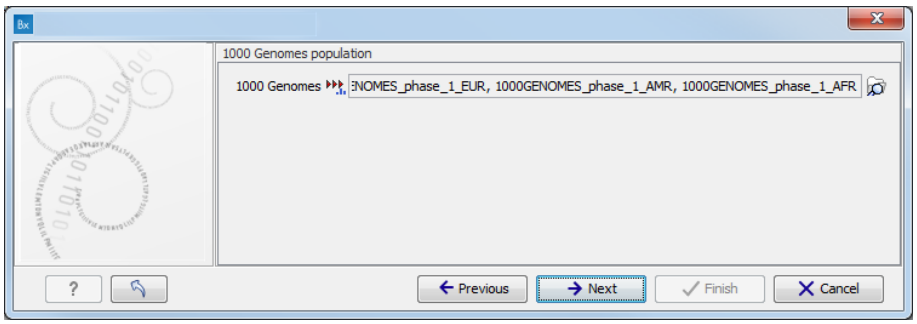


Figure 17.49: *Select the relevant 1000 Genomes population(s).*

4. Specify the **1000 Genomes population** that should be used for **filtering** out variants found in the 1000 Genomes project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

5. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 17.50).

   This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

6. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Filter Causal Variants (TAS-HD) workflow

Three types of output are generated:

- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- A **Genome Browser View**
- A **Filtered Variant Track**

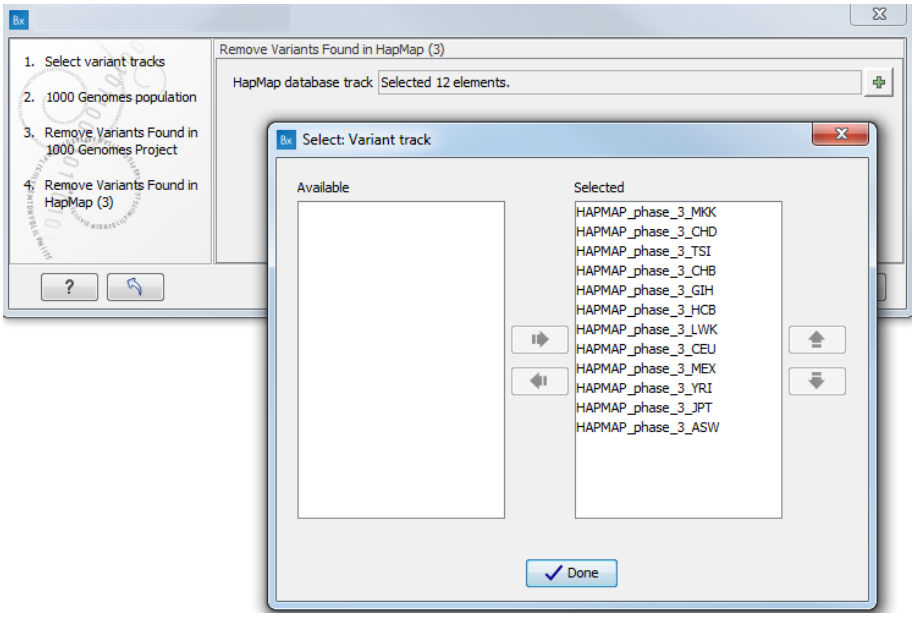


Figure 17.50: *Select the relevant Hapmap population(s).*

17.3.2 Identify Causal Inherited Variants in Family of Four (TAS)

As the name of the workflow implies, you can use the **Identify Causal Inherited Variants in a Family of Four (TAS)** ready-to-use workflow to identify inherited causal variants in a family of four. The family relationship can be a child, a mother, a father and one additional affected family member where, in addition to the child (the proband) one of the parents are affected and one additional family member is affected. The fourth family member can be any related and affected family member such as a sibling, grand parent, uncle or the like.

The **Identify Causal Inherited Variants in a Family of Four (TAS)** ready-to-use workflow accepts sequencing reads as input from each of the four family members.

How to run the Identify Causal Inherited Variants in a Family of Four (TAS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Family of Four (TAS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Causal Inherited Variants in a Family of Four (TAS)** ()

1. Double-click on the **Identify Causal Inherited Variants in a Family of Four (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 17.51).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that

should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

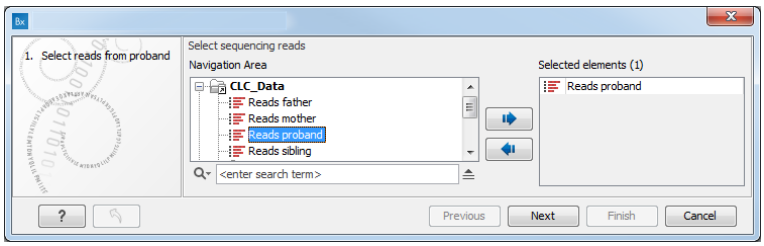


Figure 17.51: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from the **affected parent**.
4. Select the sequencing reads from the **unaffected parent**.
5. Select the sequencing reads from the **affected family member**.
6. Select the targeted region file (figure 17.52).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

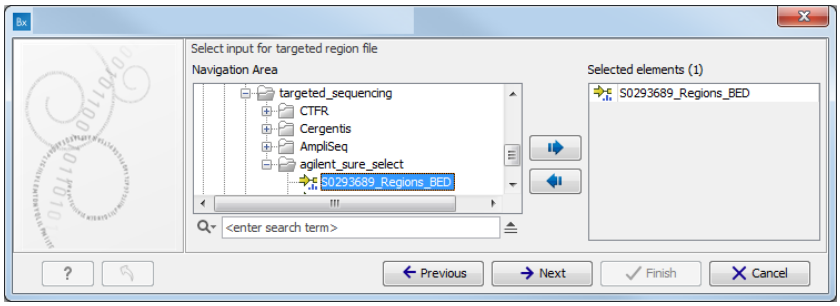


Figure 17.52: *Select the targeted region file you used for sequencing.*

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 17.53).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

   - **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability

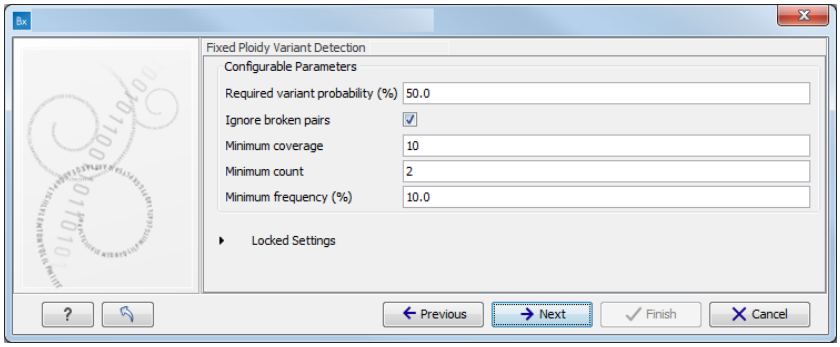


Figure 17.53: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
10. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected family member**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 17.54).

    When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

    The parameters that can be set are:

    - **Minimum coverage** provides the length of each target region that has at least this coverage.
    - **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.

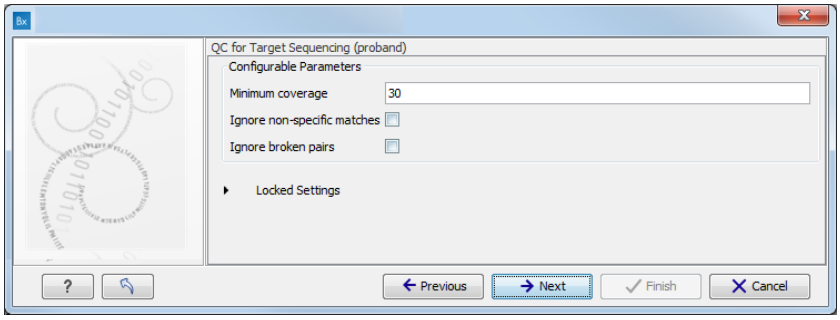


Figure 17.54: *Specify the parameters for the QC for Target Sequencing tool.*

- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

For more information about the tool, see section 20.1.

12. Specify the parameters for the **QC for Target Sequencing** tool for the **affected parent**.
13. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected parent**.
14. Specify the parameters for the **QC for Target Sequencing** tool for the **affected family member**.
15. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 17.55).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

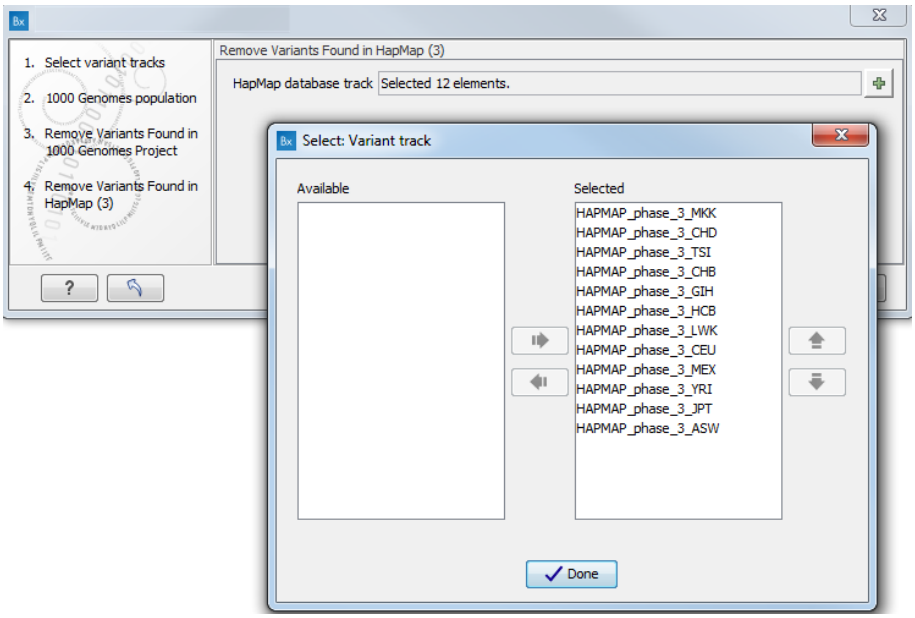


Figure 17.55: *Select the relevant Hapmap population(s).*

16. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Causal Inherited Variants in a Family of Four (TAS) workflow

Six types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

17.3.3 Identify Causal Inherited Variants in Trio (TAS)

The **Identify Causal Inherited Variants in a Trio (TAS)** ready-to-use workflow identifies putative disease causing inherited variants by creating a list of variants present in both affected individuals and subtracting all variants in the unaffected individual. The workflow includes a back-check for all family members

The **Identify Causal Inherited Variants in a Trio (TAS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Causal Inherited Variants in a Trio (TAS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Causal Inherited Variants in a Trio (TAS)** workflow, go to:

Toolbox | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Causal Inherited Variants in a Trio (TAS)** ()

1. Double-click on the **Identify Causal Inherited Variants in a Trio (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 17.56).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

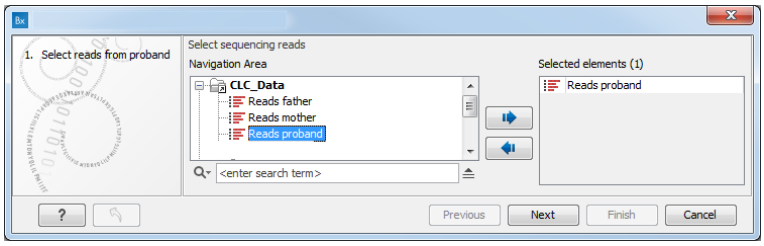


Figure 17.56: *Specify the sequencing reads for the appropriate family member.*

3. Select the reads for the **affected parent**.

4. Select the reads for the **unaffected parent**.

5. Select the targeted region file (figure 17.57).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

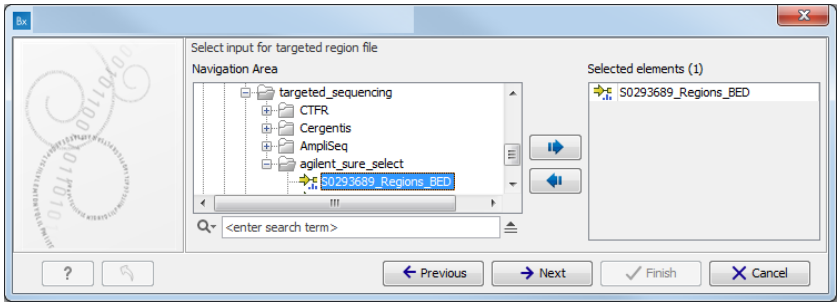


Figure 17.57: *Select the targeted region file you used for sequencing.*

6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 17.58).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

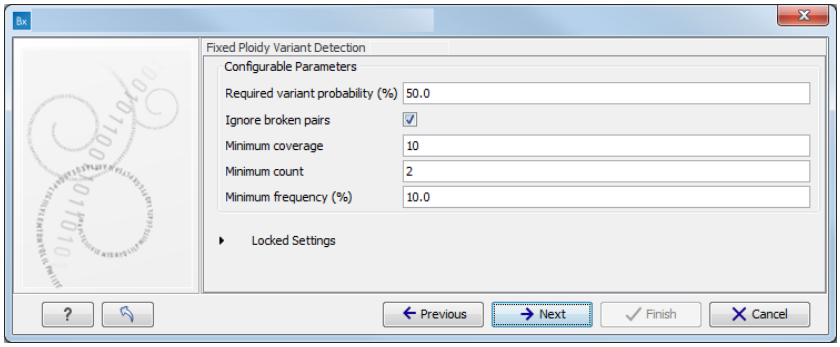


Figure 17.58: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **affected parent**.
8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected parent**.
9. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 17.59).

   When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

   The parameters that can be set are:

   - **Minimum coverage** provides the length of each target region that has at least this coverage.

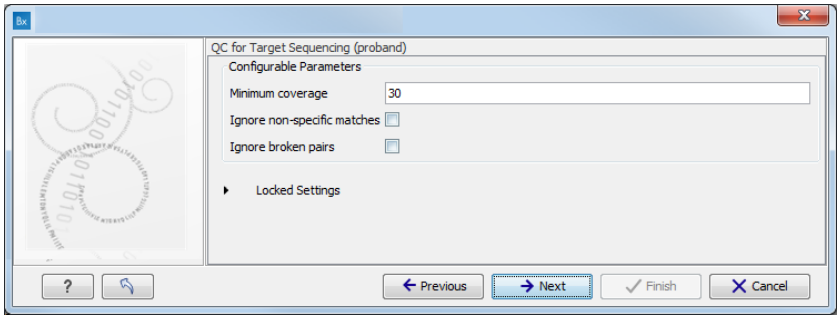


Figure 17.59: *Specify the parameters for the QC for Target Sequencing tool.*

- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

10. Specify the parameters for the **QC for Target Sequencing** tool for the **affected parent**.

11. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected parent**.

12. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap (figure 17.60).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

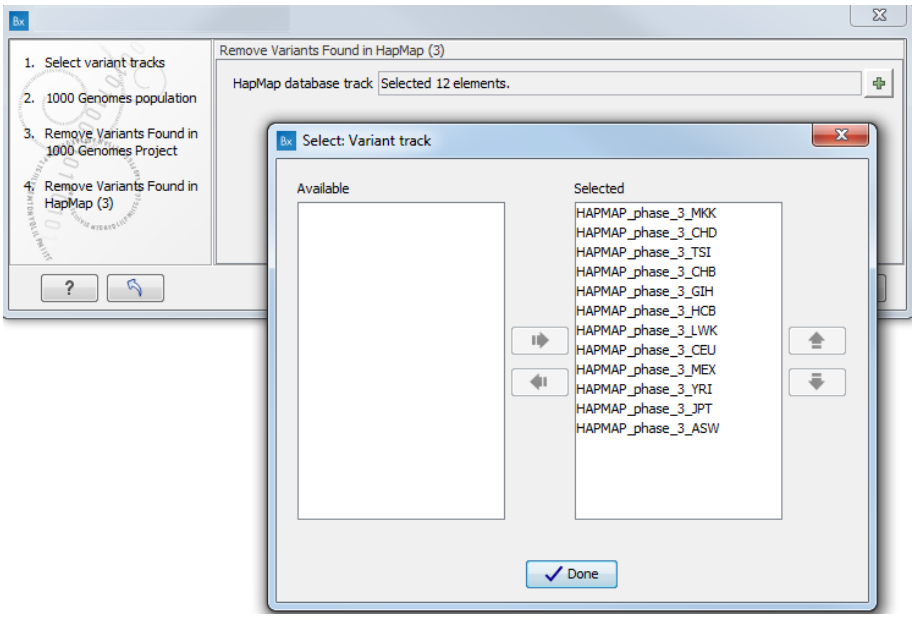


Figure 17.60: *Select the relevant Hapmap population(s).*

13. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Causal Inherited Variants in a Trio (TAS) workflow

Six types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variants in ...** One track for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Putative Causal Variants in Child** The putative disease-causing variants identified in the child. The variant track can be opened in table view to see all information about the variants.
- **Gene List with Putative Causal Variants** Gene track with the identified putative causal variants in the child. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

17.3.4 Identify Rare Disease Causing Mutations in Family of Four (TAS)

You can use the **Identify Rare Disease Causing Mutations in a Family of Four (TAS)** ready-to-use workflow to identifie de novo and compound heterozygous variants from an extended family of four, where the fourth individual is not affected.

The **Identify Rare Disease Causing Mutations in a Family of Four (TAS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Family of Four (TAS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Family of Four (TAS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Rare Disease Causing Mutations in a Family of Four (WGS** ()

1. Double-click on the **Identify Rare Disease Causing Mutations in a Family of Four (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads from the **proband** (figure 17.61).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

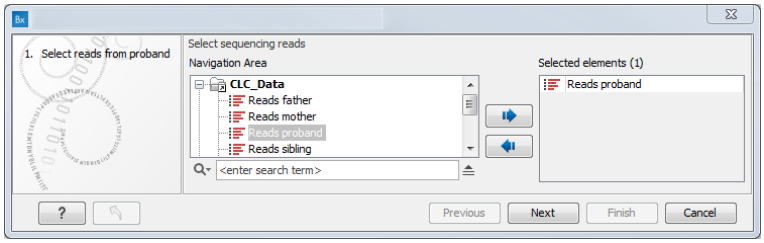


Figure 17.61: *Specify the sequencing reads for the appropriate family member.*

3. Select the sequencing reads from for the **mother**.

4. Select the sequencing reads from the **father**.

5. Select the sequencing reads from the **unaffected sibling**.

6. Select the **targeted region** file (figure 17.62).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

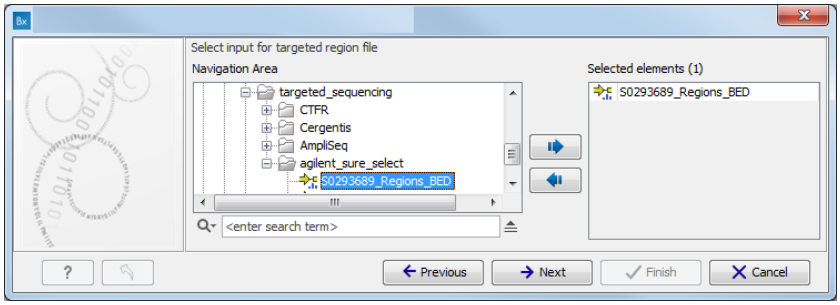


Figure 17.62: *Select the targeted region file you used for sequencing.*

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 17.63).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

   The parameters that can be set are:

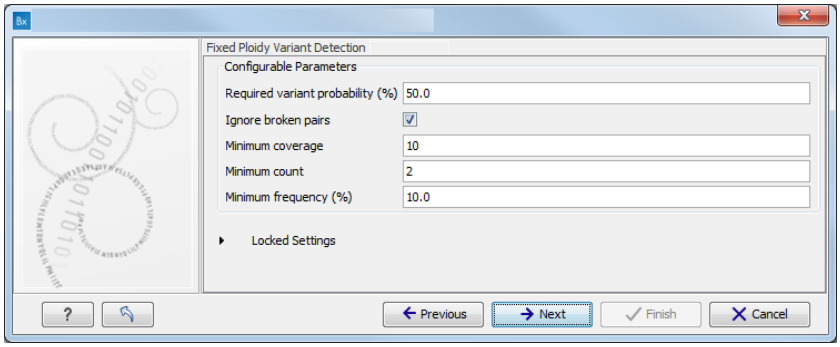


Figure 17.63: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **mother**.
9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **father**.
10. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **unaffected sibling**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 17.64).

    When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

    The parameters that can be set are:

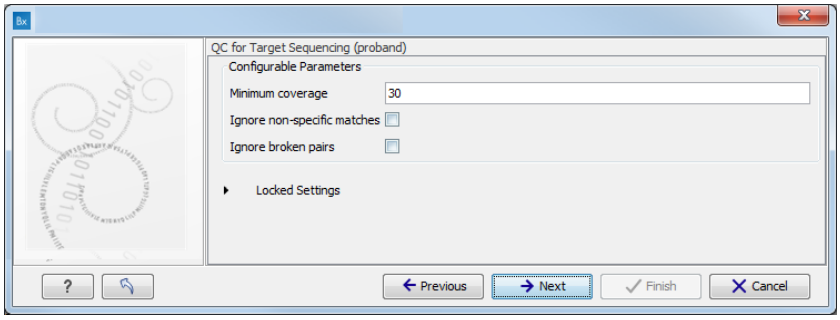


Figure 17.64: *Specify the parameters for the QC for Target Sequencing tool.*

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

12. Specify the parameters for the **QC for Target Sequencing** tool for the **mother**.
13. Specify the parameters for the **QC for Target Sequencing** tool for the **father**.
14. Specify the parameters for the **QC for Target Sequencing** tool for the **unaffected sibling**.
15. Specify the affected child's **gender** (figure 17.65).

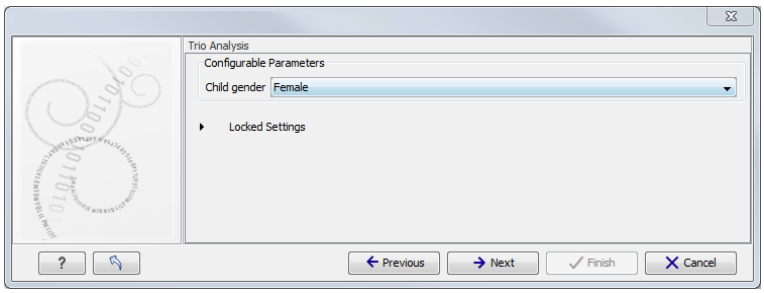


Figure 17.65: *Specify the proband's gender.*

16. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father** (figure 17.66).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

17. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **mother**.
18. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.
19. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.

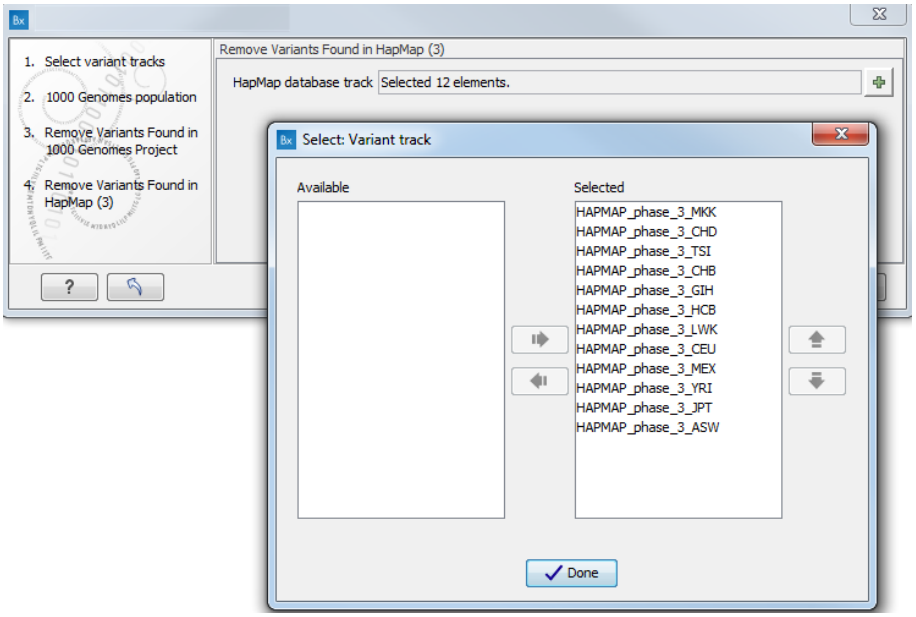


Figure 17.66: *Select the relevant Hapmap population(s).*

20. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Rare Disease Causing Mutations in a Family of Four (TAS) workflow

Twelve different types of output are generated:

- **Reads Mapping** One for each family member. The reads mapped to the reference sequence.
- **Variant Tracks** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with de novo Variants** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.

- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.
- **De novo variants** Variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.

17.3.5 Identify Rare Disease Causing Mutations in Trio (TAS)

The **Identify Rare Disease Causing Mutations in a Trio (TAS)** identifies de novo and compound heterozygous variants from a Trio. The workflow includes a back-check for all family members.

The **Identify Rare Disease Causing Mutations in a Trio (TAS)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Rare Disease Causing Mutations in a Trio (TAS) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Rare Disease Causing Mutations in a Trio (TAS)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Rare Disease Causing Mutations in a Trio (TAS** ()

1. Double-click on the **Identify Rare Disease Causing Mutations in a Trio (TAS)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.
2. Select the sequencing reads from the **proband** (figure 17.67).

   The sequencing reads from the different family members are specified one at a time in the appropriate window. The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.
3. Select the sequencing reads from the **mother**.
4. Select the sequencing reads from the **father**.

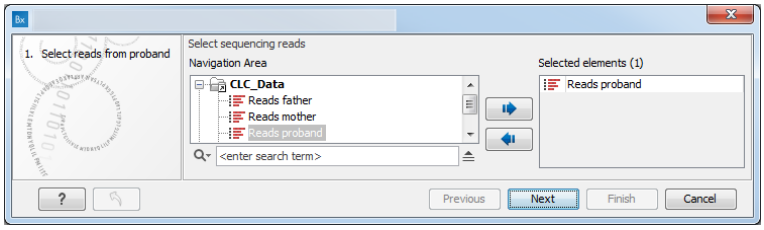


Figure 17.67: *Specify the sequencing reads for the appropriate family member.*

5. You then need to select the **targeted region** file (figure 17.68).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

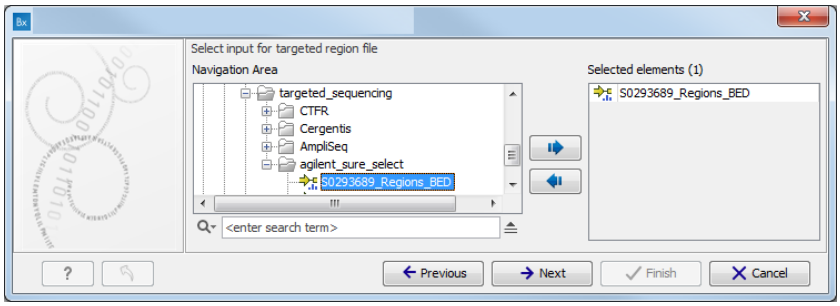


Figure 17.68: *Select the targeted region file you used for sequencing.*

6. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **proband** (figure 17.69).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

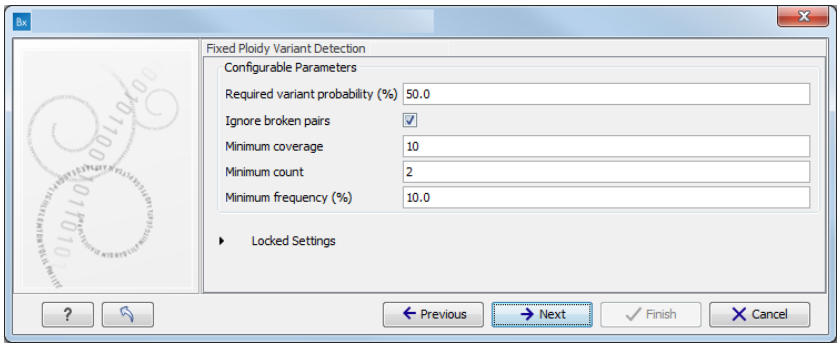


Figure 17.69: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, section 22.13.

7. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **mother**.
8. Specify the parameters for the **Fixed Ploidy Variant Detection** tool for the **father**.
9. Specify the parameters for the **QC for Target Sequencing** tool for the **proband** (figure 17.70).

   When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

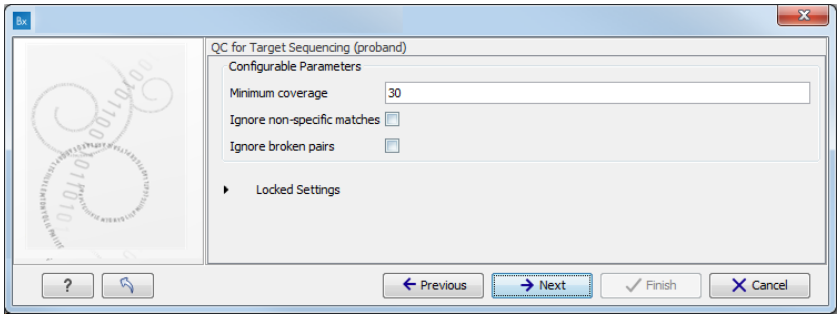


Figure 17.70: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.

- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

10. Specify the parameters for the **QC for Target Sequencing** tool for the **mother**.
11. Specify the parameters for the **QC for Target Sequencing** tool for the **father**.
12. Specify the affected child's gender for the Trio analysis (figure 17.71).

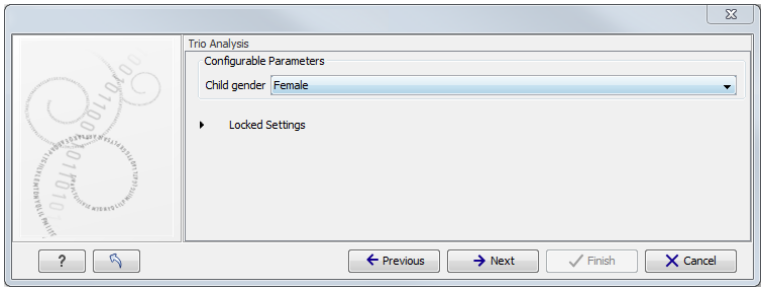


Figure 17.71: *Specify the proband's gender.*

13. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **father** (figure 17.72).

    This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

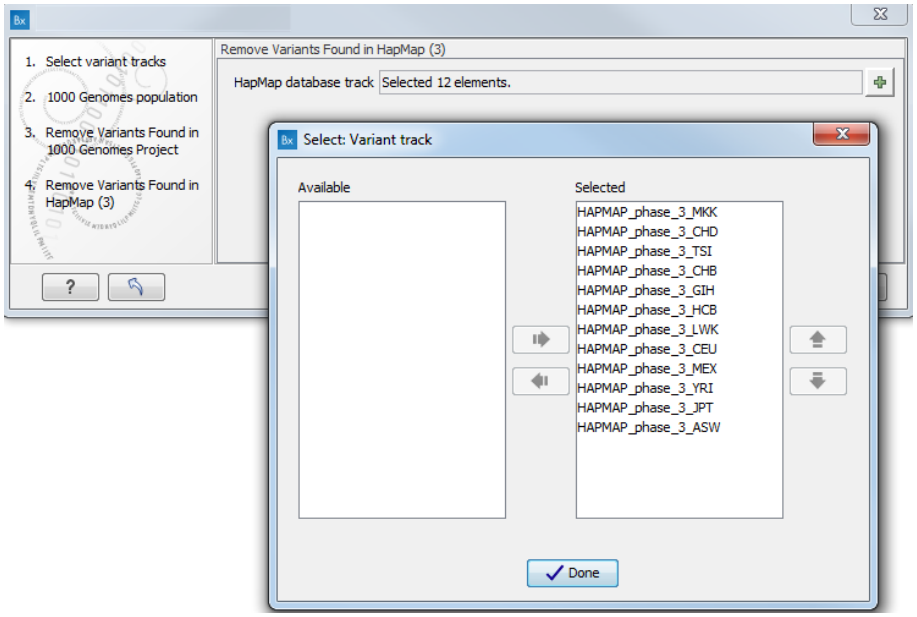


Figure 17.72: *Select the relevant Hapmap population(s).*

14. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap for the **mother**.
15. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **de novo variants**.

16. Specify the **Hapmap populations** that should be used for **filtering out** variants found in Hapmap from the **recessive variants**.

17. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters and it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Rare Disease Causing Mutations in a Trio (TAS) workflow

Twelve different types of output are generated:

- **Reads Tracks** One for each family member. The reads mapped to the reference sequence.
- **Variant Tracks** One for each family member. The variants identified in each of the family members. The variant track can be opened in table view to see all information about the variants.
- **De novo variants** Variant track showing de novo variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Recessive variants** Variant track showing recessive variants in the proband. The variant track can be opened in table view to see all information about the variants.
- **Identified Compound Heterozygous Genes Proband** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with de novo Variants** Gene track with the identified putative compound heterozygous Variants in the proband. The gene track can be opened in table view to see the gene names.
- **Gene List with recessive Variants** Gene track with the identified recessive variants in the proband. The gene track can be opened in table view to see the gene names.
- **Target Region Coverage Report** One for each family member. The report consists of a number of tables and graphs that in different ways provide information about the mapped reads from each sample.
- **Target Region Coverage** One track for each individual. When opened in table format, it is possible to see a range of different information about the targeted regions, such as target region length, read count, and base count.
- **Genome Browser View** This is a collection of tracks shown together in a view that makes it easy to compare information from the individual tracks, such as compare the identified variants with the read mappings and information from databases.
- **De novo Mutations Amino Acid Track**
- **Recessive Variants Amino Acid Track**

17.3.6 Identify Variants (TAS-HD)

You can use the **Identify Variants (TAS-HD)** ready-to-use workflow to call variants in the mapped and locally realigned reads. The workflow removes false positives and, in case of a targeted experiment, removes variants outside the targeted region. Variant calling is performed with the Fixed Ploidy Variant Detection tool.

The **Identify Variants (TA-HD)** ready-to-use workflow accepts sequencing reads as input.

How to run the Identify Variants (TAS-HD) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify Variants (TAS-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify Variants (TAS-HD** ()

1. Double-click on the **Identify Variants (TAS-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads you want to analyze (figure 17.73). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

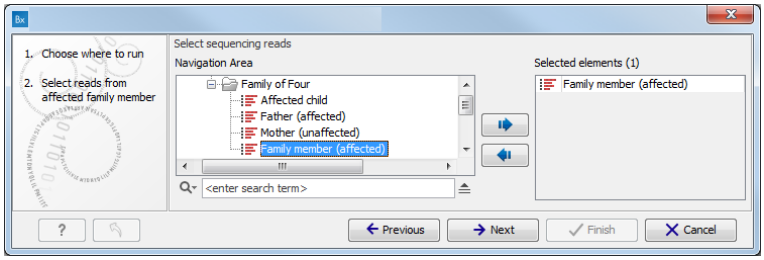


Figure 17.73: *Specify the sequencing reads for the appropriate family member.*

3. Specify a target region file for the **Indels and Structural Variants** tool (figure 17.74).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

4. Specify the parameters for the **Fixed Ploidy Variant Detection** tool, including a target region file (figure 17.75).

   The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

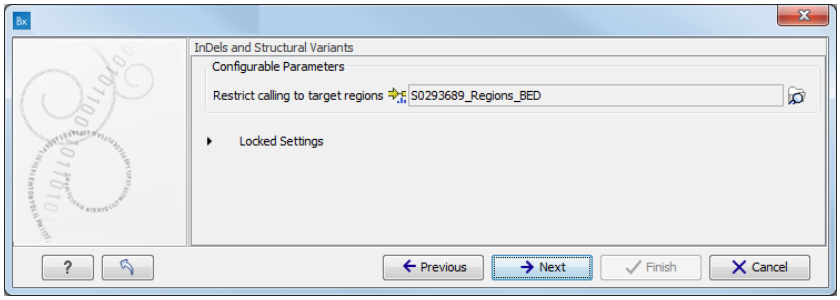


Figure 17.74: *Specify the parameters for the Indels and Structural Variants tool.*

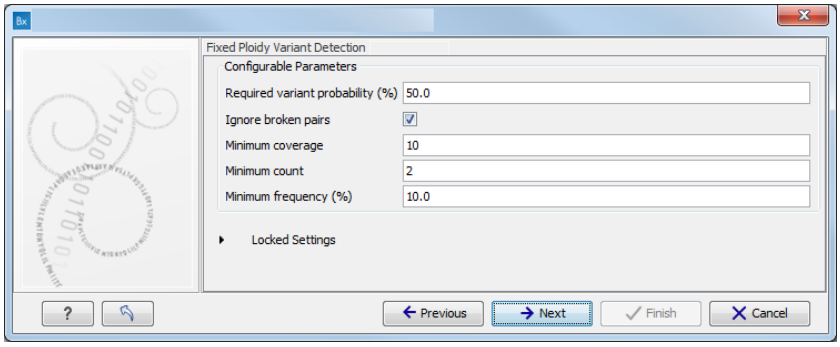


Figure 17.75: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.
- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

5. Specify the parameters for the **QC for Target Sequencing** tool, including a target region file (figure 17.76).

   When working with targeted data (WES or TAS data), quality checks for the targeted sequencing is included in the workflows. Again, you can choose to use the default settings, or you can choose to adjust the parameters.

   The parameters that can be set are:

   - **Minimum coverage** provides the length of each target region that has at least this coverage.
   - **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
   - **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

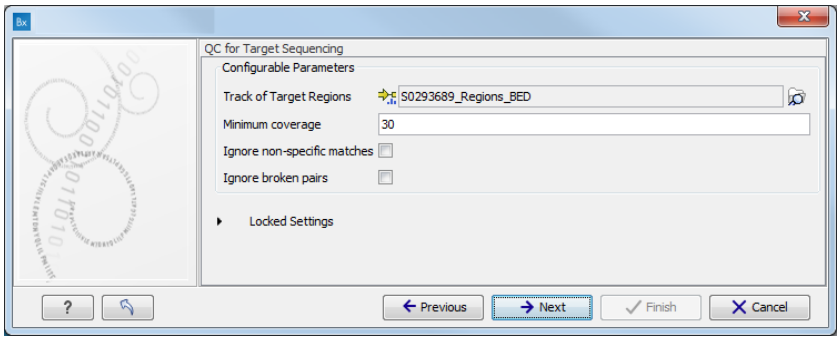


Figure 17.76: *Specify the parameters for the QC for Target Sequencing tool.*

   For more information about the tool, see section 20.1.

6. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify Variants (TAS-HD) workflow

Four types of output are generated:

- A **Reads Track** Read Mapping
- A **Filtered Variant Track** Identified variants
- A **Coverage Report**
- A **Per-region Statistics Track**

17.3.7 Identify and Annotate Variants (TAS-HD)

The **Identify and Annotate Variants (TAS-HD)** tool should be used to identify and annotate variants in one sample. The tool consists of a workflow that is a combination of the **Identify Variants** and the **Annotate Variants** workflows.

The tool runs an internal workflow, which starts with mapping the sequencing reads to the human reference sequence. Then it runs a local realignment to improve the variant detection,

which is run afterwards. After the variants have been detected, they are annotated with gene names, amino acid changes, conservation scores, information from relevant variants present in the ClinVar database, and information from common variants present in the common dbSNP, HapMap, and 1000 Genomes database. Furthermore, a targeted region report is created to inspect the overall coverage and mapping specificity.

The difference between Identify and Annotate Variants (TAS-HD) and (WES-HD) is that the **Autodetect paired distances** has been switched off in Map Reads to Reference tool for the TAS workflows.

How to run the Identify and Annotate Variants (TAS-HD) workflow

This section recapitulates the steps you need to take to start the workflow, each item corresponding to a different wizard windows. For more information on the specific tools used in this workflow, see section 12.3.

To run the **Identify and Annotate Variants (TAS-HD)** workflow, go to:

> **Toolbox** | **Ready-to-Use Workflows** | **Targeted Amplicon Sequencing Sequencing** () | **Hereditary Disease** () | **Identify and Annotate Variants** ()

1. Double-click on the **Identify and Annotate Variants (TAS-HD)** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the sequencing reads you want to analyze (figure 17.77). The panel in the left side of the wizard shows the kind of input that should be provided. Select by double-clicking on the reads file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard.

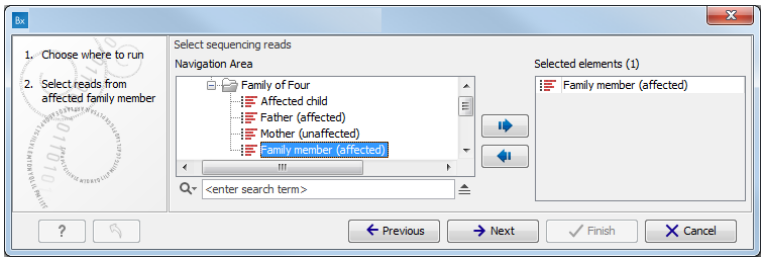


Figure 17.77: *Specify the sequencing reads for the appropriate family member.*

3. Specify which 1000 Genomes population you would like to use (figure 17.78).

4. Specify a target region file for the **Indels and Structural Variants** tool. (figure 17.79).

   The targeted region file is a file that specifies which regions have been sequenced, when working with whole exome sequencing or targeted amplicon sequencing data. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents.

5. Specify the **Fixed Ploidy Variant Detection** settings, including a target region file (figure 17.80).

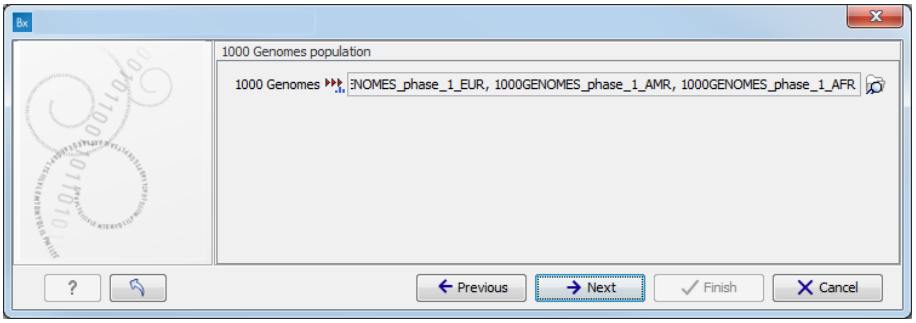


Figure 17.78: *Select the relevant 1000 Genomes population(s).*

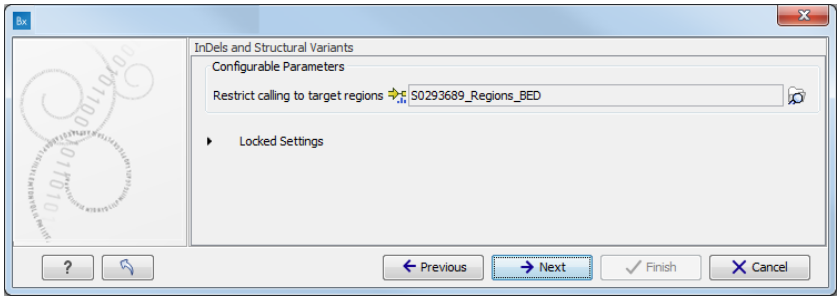


Figure 17.79: *Specify the parameters for the Indels and Structural Variants tool.*

The parameters used by the Fixed Ploidy Variant Detection tool can be adjusted. We have optimized the parameters to the individual analyses, but you may want to tweak some of the parameters to fit your particular sequencing data. A good starting point could be to run an analysis with the default settings.

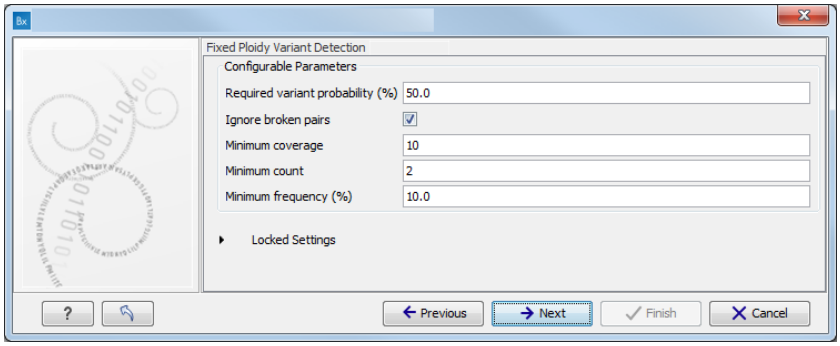


Figure 17.80: *Specify the parameters for the Fixed Ploidy Variant Detection tool.*

The parameters that can be set are:

- **Required variant probability** is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater

than 0.9.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

For more information about the tool, see section 22.13.

6. Specify the parameters for the **QC for Target Sequencing** tool, including a target region file (figure 17.81).

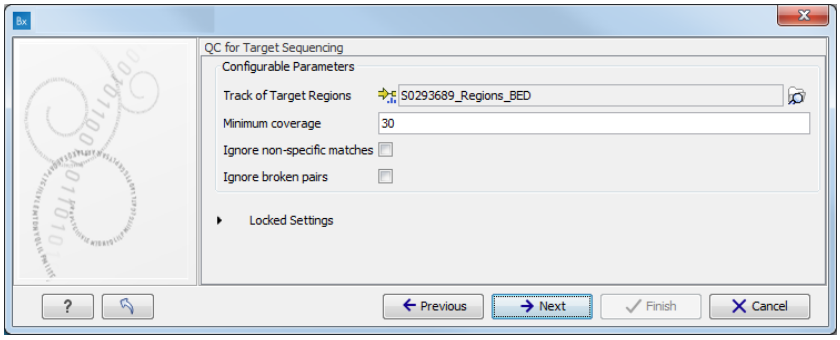


Figure 17.81: *Specify the parameters for the QC for Target Sequencing tool.*

The parameters that can be set are:

- **Minimum coverage** provides the length of each target region that has at least this coverage.
- **Ignore non-specific matches**: reads that are non-specifically mapped will be ignored.
- **Ignore broken pairs**: reads that belong to broken pairs will be ignored.

For more information about the tool, see section 20.1.

7. Specify a **targeted region** file to remove variants outside of this region. (figure 17.82)

8. Specify the 1000 Genomes population that should be used to add information on variants found in the 1000 Genomes project. This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

9. Specify the Hapmap population that should be used to add information on variants found in the Hapmap project. This can be done using the drop-down list found in this wizard step.

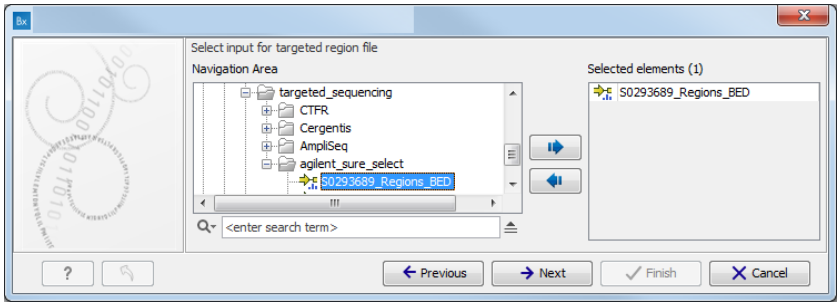


Figure 17.82: *Select the targeted region file you used for sequencing.*

Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

10. Pressing the button **Preview All Parameters** allows you to preview all parameters. At this step you can only view the parameters, it is not possible to make any changes. Choose to save the results and click on the button labeled **Finish**.

Output from the Identify and Annotate Variants (TAS-HD) workflow

Six types of output are generated:

- A **Reads Track**
- A **Coverage Report Read Mapping**
- A **Per-region Statistics Track**
- A **Filtered Variant Track** Annotated variants
- An **Amino Acid Track** Shows the consequences of the variants at the amino acid level in the context of the original amino acid sequence. A variant introducing a stop mutation is illustrated with a red amino acid.
- A **Genome Browser View**

Chapter 18

Whole Transcriptome Sequencing (WTS)

Contents

The technologies originally developed for next-generation DNA sequencing can also be applied to deep sequencing of the transcriptome. This is done through cDNA sequencing and is called RNA sequencing or simply RNA-seq.

One of the key advantages of RNA-seq is that the method is independent of prior knowledge of the corresponding genomic sequences and therefore can be used to identify transcripts from unannotated genes, novel splicing isoforms, and gene-fusion transcripts [Wang et al., 2009, Martin and Wang, 2011]. Another strength is that it opens up for studies of transcriptomic complexities such as deciphering allele-specific transcription by the use of SNPs present in the transcribed regions [Heap et al., 2010].

RNA-seq-based transcriptomic studies have the potential to increase the overall understanding of the transcriptome. However, the key to get access to the hidden information and be able to make a meaningful interpretation of the sequencing data highly relies on the downstream bioinformatic analysis.

In this chapter we will first discuss the initial steps in the data analysis that lie upstream of the analysis using ready-to-use workflows. Next, we will look at what the individual ready-to-use workflows can be used for and go through step by step how to run the workflows.

The *Biomedical Genomics Workbench* offers a range of different tools for RNA-seq analysis. Currently 5 different ready-to-use workflows for 3 different species (**human** (), **mouse** () and **rat** ()) are available for analysis of RNA-seq data:

- Annotate Variants (WTS)
- Compare Variants in DNA and RNA

- Identify Candidate Variants and Genes from Tumor Normal Pair
- Identify Variants and Add Expression Values
- Identify and Annotate Differentially Expressed Genes and Pathways

The ready-to-use workflows can be found in the toolbox under Whole Transcriptome Sequencing as shown in figure 18.1.

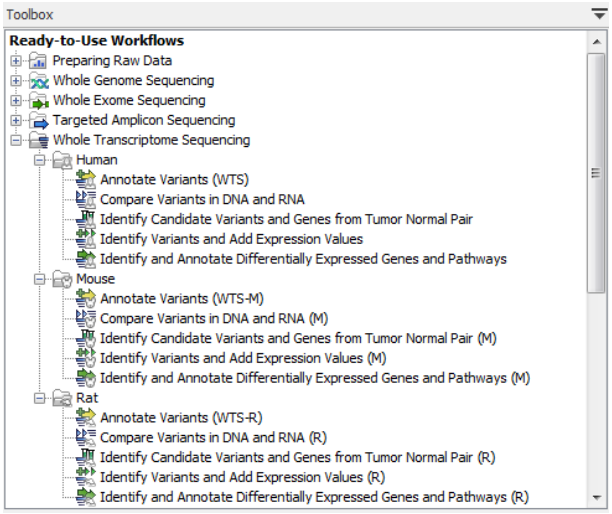


Figure 18.1: *The RNA-seq ready-to-use workflows.*

Note! Often you will have to prepare data with one of the two **Preparing Raw Data** workflows described in section 14 before you proceed to the analysis of the sequencing data **RNA-Seq**.

Note! Make sure that you have selected the references corresponding to the species you will be working with. To check and potentially change which Reference Data Set is currently in use, click on the **Data Management** ( ) button in the top right corner of the Workbench, and click apply to the appropriate data set (Hg38, Hg19, Mouse or Rat). If you are given an error message about missing a reference data element when starting a workflow, you can delete and re-download the missing reference element or set.

Also note that in case of workflows annotating variants using databases available for more than one population, you can select the population that matches best the population your samples are derived from. This will be done in the wizard for populations from the 1000 Genomes Project, while Hapmap populations can be specified with the **Data Management** ( ) function before starting the workflows (see section 13.1).

18.1 Analysis of multiple samples

To get the most out of the RNA-Seq analysis tools of the workbench, we recommend that all input expression tracks have associated metadata. For information about how to use and setup metadata, please see section 29.

18.2 Annotate Variants (WTS)

Using a variant track ( ) (e.g. the output from the Identify Variants and Add Expression Values ready-to-use workflow) the **Annotate Variants (WTS)** ready-to-use workflow runs an "internal" workflow that adds the following annotations to the variant track:

- **Gene names** Adds names of genes whenever a variant is found within a known gene.
- **mRNA** Adds names of mRNA whenever a variant is found within a known transcript.
- **CDS** Adds names of CDS whenever a variant is found within a coding sequence.
- **Amino acid changes** Adds information about amino acid changes caused by the variants.
- **Information from ClinVar** Adds information about the relationships between human variations and their clinical significance.
- **Information from dbSNP** Adds information from the "Single Nucleotide Polymorphism Database", which is a general catalog of genome variation, including SNPs, multinucleotide polymorphisms (MNPs), insertions and deletions (indels), and short tandem repeats (STRs).
- **PhastCons Conservation scores** The conservation scores, in this case generated from a multiple alignment with a number of vertebrates, describe the level of nucleotide conservation in the region around each variant.

1. Go to the toolbox and select the **Annotate Variants (WTS)** workflow. In the first wizard step, select the input variant track (figure 18.2).

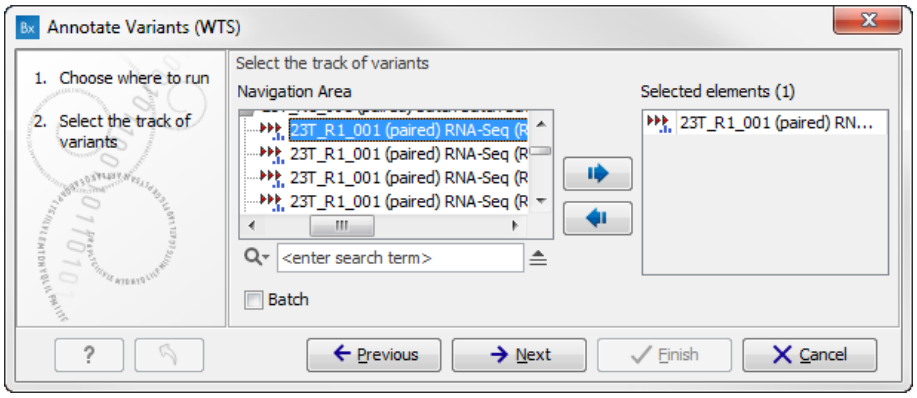


Figure 18.2: *Select the variant track to annotate.*

2. Click on the button labeled **Next**. If you are using the workflow from the Human folder, you should specify which 1000 Genomes population yo use (figure 18.3). This can be done using the drop-down list found in this wizard step. Please note that the populations available from the drop-down list can be specified with the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

3. Click on the button labeled **Next** to go to the last wizard step (figure 18.4).

   In this wizard step you can check the selected settings by clicking on the button labeled **Preview All Parameters**. In the **Preview All Parameters** wizard you can only check the settings, it is not possible to make any changes at this point.

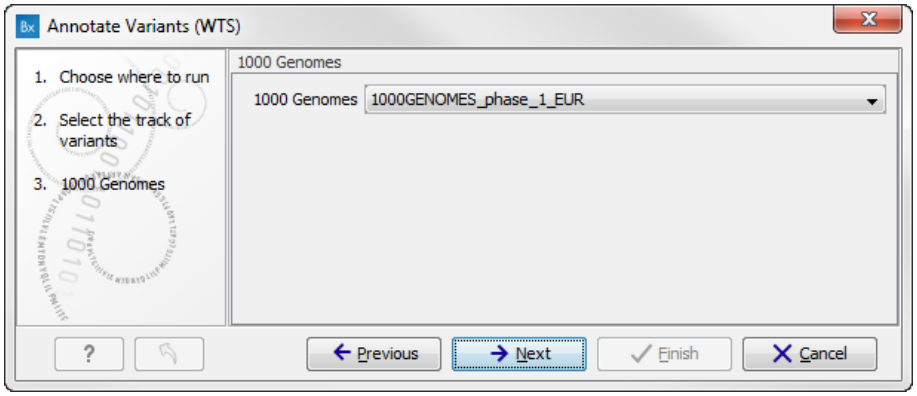


Figure 18.3: *Select the relevant 1000 Genomes population(s).*

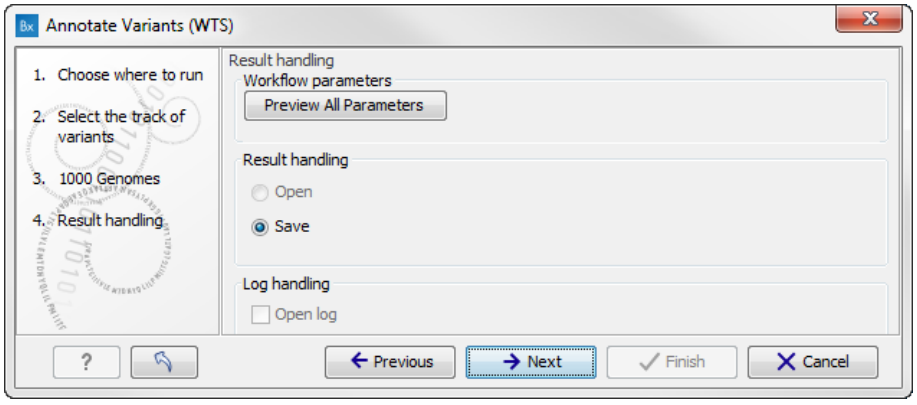


Figure 18.4: *Check the settings and save your results.*

4. Choose to **Save** your results and click on the button labeled **Finish**.

Two types of output are generated:

1. **Annotated Variants** ( ) Annotation track showing the variants. Hold the mouse over one of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant.

2. **Genome Browser View Annotated Variants** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in dbSNP, ClinVar, 1000 Genomes, and PhastCons conservation scores (see figure 18.5).

Note! Please be aware that if you delete the annotated variant track, this track will also disappear from the genome browser view.

It is possible to add tracks to the Genome Browser View such as mapped sequencing reads as well as other tracks. This can be done by dragging the track directly from the **Navigation Area** to the Genome Browser View.

If you double-click on the name of the annotated variant track in the left hand side of the Genome Browser View, a table that includes all variants and the added information/annotations will open

Figure 18.5: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list) containing individual tracks for all added annotations.*

(see figure 18.6). The table and the Genome Browser View are linked; if you click on an entry in the table, this particular position in the genome will automatically be brought into focus in the Genome Browser View.

You may be met with a warning as shown in figure 18.7. This is simply a warning telling you that it may take some time to create the table if you are working with tracks containing large amounts of annotations. Please note that in case none of the variants are present in ClinVar or dbSNP, the corresponding annotation column headers are missing from the result.

Adding information from other sources may help you identify interesting candidate variants for further research. E.g. known common genetic variants (present in the HapMap database) or variants known to play a role in drug response or other relevant phenotypes (present in the ClinVar database) can easily be identified. Further, variants not found in the ClinVar database,

Figure 18.6: *The output from the Annotate Variants ready-to-use workflow is a genome browser view (a track list). The information is also available in table view. Click on the small table icon to open the table view. If you hold down the "Ctrl" key while clicking on the table icon, you will open a split view showing both the genome browser view and the table view.*

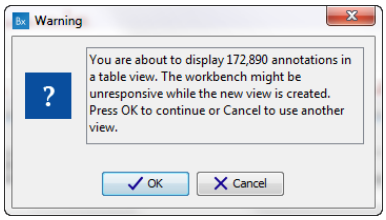


Figure 18.7: *Warning that appears when you work with tracks containing many annotations.*

can be prioritized based on amino acid changes in case the variant causes changes on the amino acid level.

A high conservation level between different vertebrates or mammals, in the region containing the variant, can also be used to give a hint about whether a given variant is found in a region with an important functional role. If you would like to use the conservation scores to identify interesting variants, we recommend that variants with a conservation score of more than 0.9 (PhastCons score) is prioritized over variants with lower conservation scores.

It is possible to filter variants based on their annotations. This type of filtering can be facilitated using the table filter found at the top part of the table. If you are performing multiple experiments where you would like to use the exact same filter criteria, you can create a filter that can be saved and reused. To do this:

Toolbox | **Identify Candidate Variants** ( ) | **Create Filter Criteria** ( )

This tool can be used to specify the filter and the **Annotate Variants** workflow should be extended by the **Identify Candidate Tool** (configured with the Filter Criterion).

Note! Sometimes the databases (e.g. dbSNP) are updated with a newer version, or maybe you have your own version of the database. In such cases you may wish to change one of the used databases. This can be done with "Data Management" function, which is described in section 13.1.

18.3 Compare variants in DNA and RNA

Integrated analysis of genomic and transcriptomic sequencing data is a powerful tool that can help increase our current understanding of genomic variants. The **Compare variants in DNA and RNA** ready-to-use workflow identifies variants in DNA and RNA and studies the relationship between the identified genomic and transcriptomic variants.

To run the ready-to-use workflow:

Toolbox | **Ready-to-Use Workflows** | **Whole Transcriptome Sequencing** ( ) | (**Human**, **Mouse** or **Rat**) | **Compare variants in DNA and RNA** ( )

1. Double-click on the **Compare variants in DNA and RNA** ready-to-use workflow to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Select the **RNA reads** that you would like to analyze (figure 18.8). Click Next.

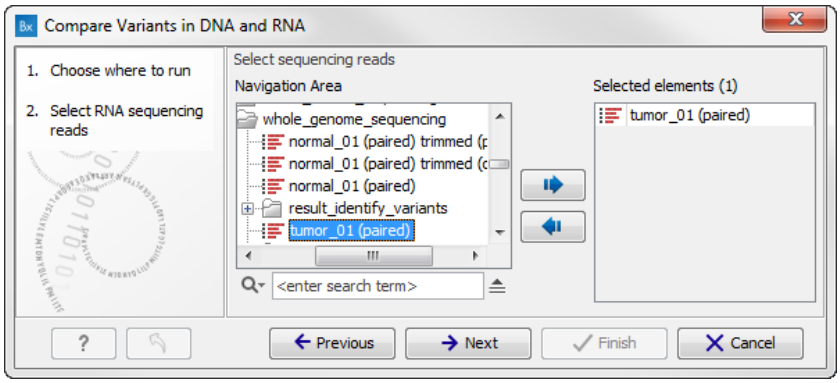


Figure 18.8: *Select the RNA reads to analyze.*

3. Select now the **DNA reads** to analyze (see figure 18.9). Click Next.

4. Configure the parameters for the RNA-Seq Analysis (figure 18.10).

   If you wish to use spike-in controls, add the relevant file in the "Spike-in controls" field.

   You can also specify that the reads should be mapped only in their forward or reverse orientation (it is by default set to both). Choosing to restrict mapping to one direction is typically appropriate when a strand specific protocol for read generation has been used, as it allows assignment of the reads to the right gene in cases where overlapping genes are located on different strands. Also, applying the 'strand specific' 'reverse' option in an

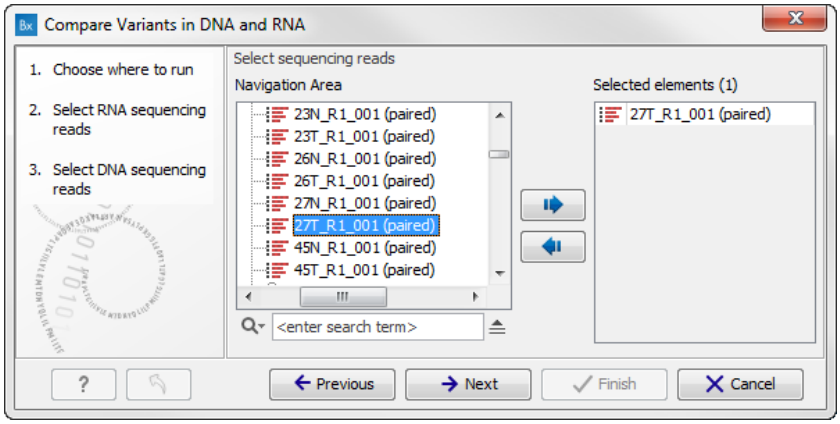


Figure 18.9: *Select the DNA reads to analyze.*

RNA-seq run could allow the user to assess the degree of antisense transcription. Note that mate pairs are not supported when choosing the forward only or reverse only option.

Click **Next**.

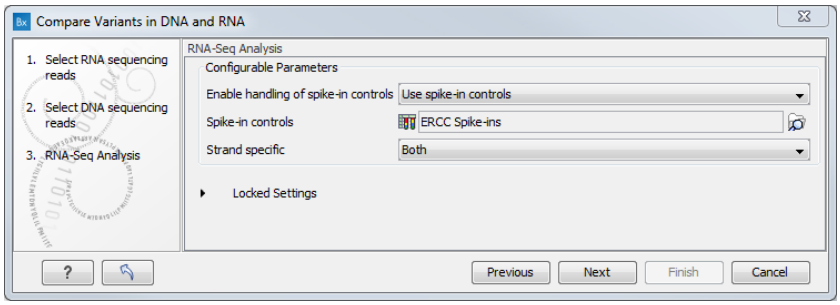


Figure 18.10: *Configure the RNA-Seq Analysis. Here we specified a file for spike-in control but left the strand specific parameter to its default value.*

5. Specify a **target region** for the analysis of the **RNA** sample with the **Indels and Structural Variants** tool (figure 18.11). Repeat for the **DNA** sample at the next step.

   The targeted region file is a file that specifies which regions have been sequenced. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents. Remember that you have a hg38-specific BED file when using hg38 as reference, and hg19-specific BED file when using hg19 as reference.

6. Set the parameters for the **Low Frequency Variant Detection** step for your **RNA** sample (see figure 18.12), and for the **DNA** sample at the next step. For a description of the different parameters that can be adjusted in the variant detection step, see section 22.14.

7. If you are working with the workflow from the Human folder, it is possible to specify in the next two steps the **1000 Genomes population** that describes best your samples (see figure 18.13). Note that this step is done twice specifying the same population(s), as we annotate first the track that will contain all variants found (Union), and then the track that will contain variants that are shared between DNA and RNA (Intersection).

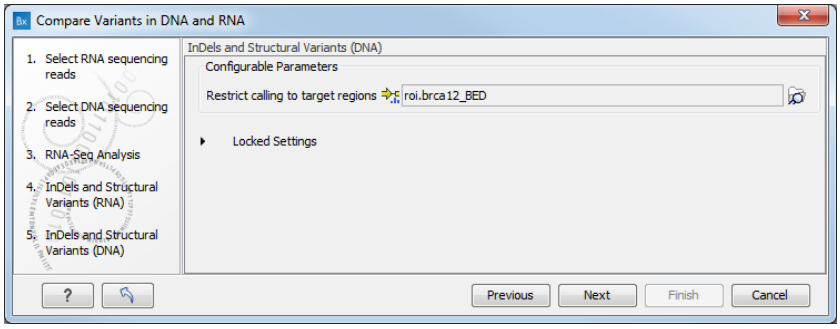


Figure 18.11: *Specify the target region for the Indels and Structural Variants tool.*

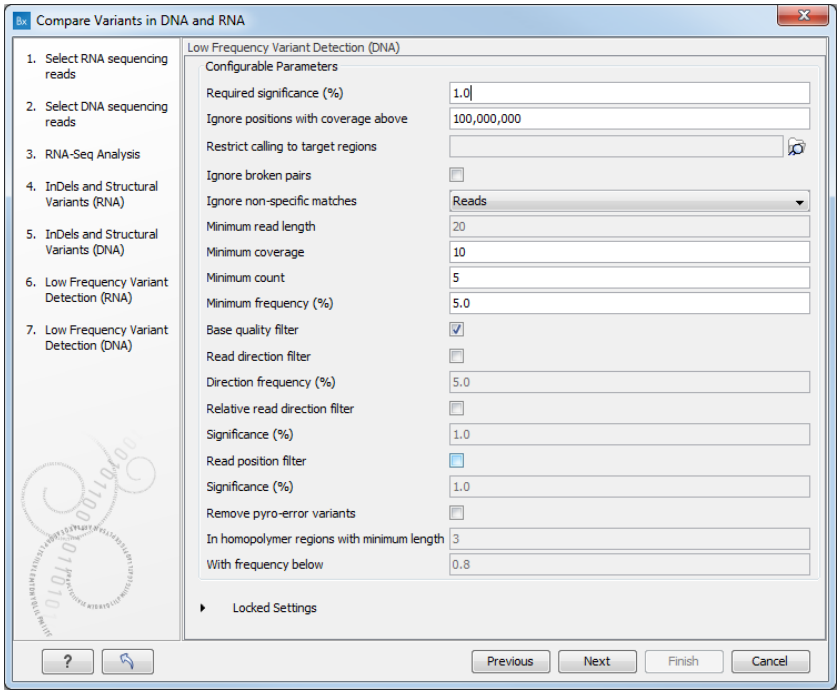


Figure 18.12: *Specify the parametes for transcriptomic variant detection.*

Under "Locked settings" you can see that "Automatically join adjacent MNVs and SNVs" has been selected. The reason for this is that many databases do not report a succession of SNVs as one MNV as is the case for the *Biomedical Genomics Workbench*, and as a consequence it is not possible to directly compare variants called with *Biomedical Genomics Workbench* with these databases. In order to support filtering against these databases anyway, the option to **Automatically join adjacent MNVs and SNVs** is enabled. This means that an MNV in the experimental data will get an exact match, if a set of SNVs and MNVs in the database can be combined to provide the same allele.

Note: This assumes that SNVs and MNVs in the track of known variants represent the same allele, although there is no evidence for this in the track of known variants.

8. Repeat the previous steps to specify the Hapmap population that characterizes best your samples. Note that this step is done twice specifying the same population, as we annotate first the track that will contain all variants found (Union), and then the track that will contain variants that are shared between DNA and RNA (Intersection).

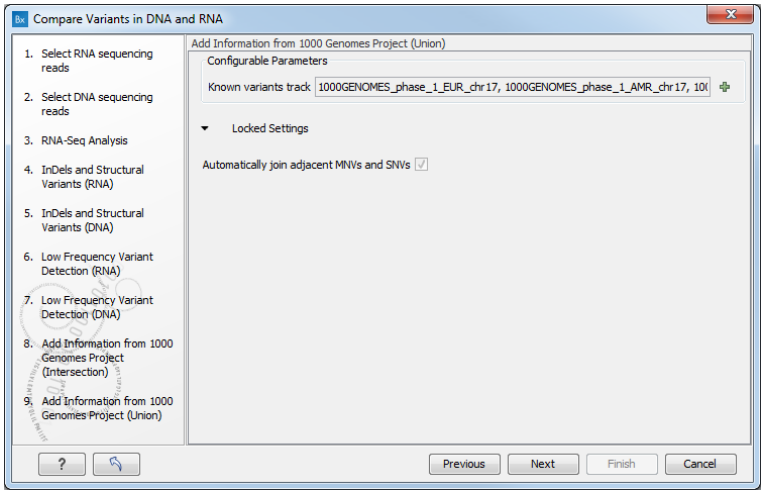


Figure 18.13: *Select the relevant population from the drop-down list.*

9. Click **Next** to go to the result handling step. **Preview All Parameters** allows you to view all parameters, but not edit them. Choose to save the results and click **Finish** to select a location to save the results and start the analysis.

Nine different output are generated:

1. A **DNA Read Mapping** and a **RNA Read Mapping** ( ) The mapped DNA or RNA sequencing reads. The sequencing reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously. For the color codes please see the description in (see section **??**).

2. A **DNA Mapping Report** and a **RNA Mapping Report** ( ) This report contains information about the reads, reference, transcripts, and statistics (see section 29.1.8 for details).

3. An **RNA Gene Expression** ( ) A track showing gene expression annotations. Hold the mouse over or right-clicking on the track. If you have zoomed in to nucleotide level, a tooltip will appear with information about gene name and expression values.

4. An **RNA Transcript Expression** ( ) A track showing transcript expression annotations. Hold the mouse over or right-clicking on the track. A tooltip will appear with information about e.g. gene name and expression values.

5. A **Filtered Variant Track with All Variants Found in DNA or RNA** ( ) This track shows all variants that have been detected in either RNA, DNA or both.

6. A **Filtered Variant Track with Variants Found in Both DNA and RNA** ( ) This track shows only the variants that are present in both DNA and RNA. With the table icon ( ) found in the lower left part of the **View Area** it is possible to switch to table view. The table view provides details about the variants such as type, zygosity, and information from a range of different databases.

7. A **Genome Browser View Variants Found in DNA and RNA** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar and dbSNP (see figure 18.14).

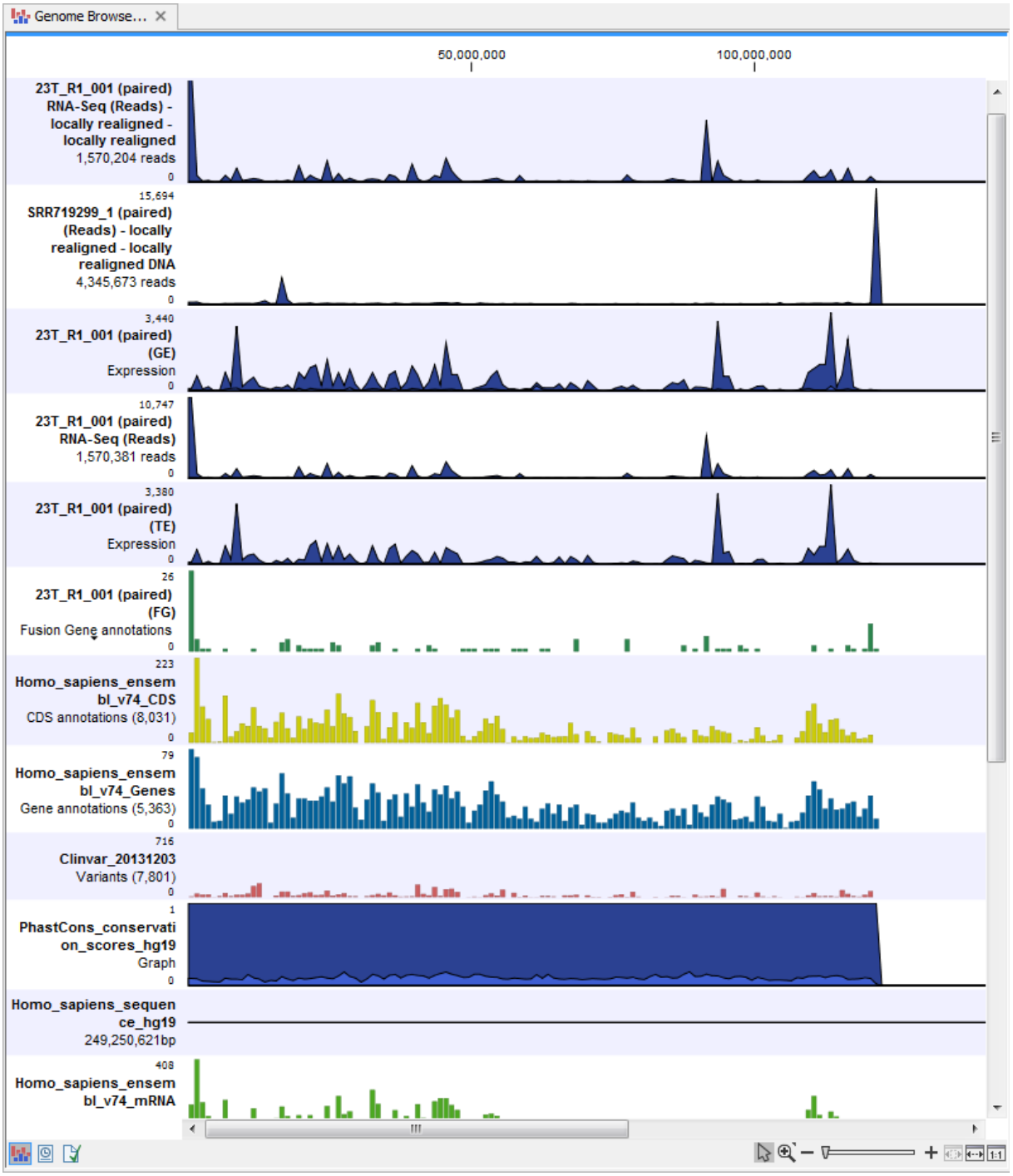


Figure 18.14: *The genome browser view makes it easy to compare a range of different data.*

The three most important tracks generated are the **Variants found in both DNA and RNA track**, **All variants found in DNA or RNA track**, and the **Genome Browser View**. The Genome Browser View makes it easy to get an overview in the context of a reference sequence, and compare variant and expression tracks with information from different databases. The two other tracks (**Variants found in both DNA and RNA track** and **All variants found in DNA or RNA track**) provides detailed information about the detected variants when opened in table view.

18.4 Identify Candidate Variants and Genes from Tumor Normal Pair

The **Identify Candidate Variants and Genes from Tumor Normal Pair** tool identifies somatic variants and differentially expressed genes in a tumor normal pair. One tumor normal pair can be compared at the time. If you would like to compare more than one pair you must repeat the analysis with the next tumor normal pair.

To run the ready-to-use workflow:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Transcriptome Sequencing** ( ) | **Human**, **Mouse** or **Rat** | **Identify Candidate Variants and Genes from Tumor Normal Pair** ( )

1. Double-click on the **Identify Candidate Variants and Genes from Tumor Normal Pair** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Specify the RNA-seq reads from the tumor sample (the panel in the left side of the wizard shows the kind of input that should be provided as in figure 18.15). Click **Next**.

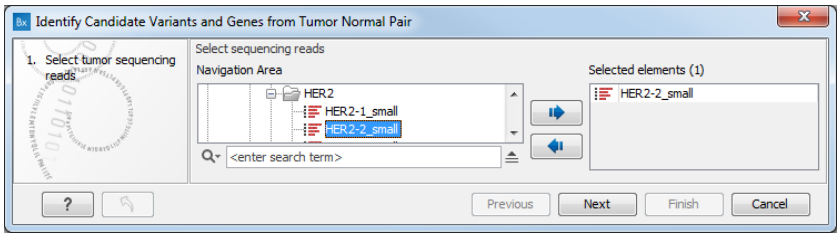


Figure 18.15: *Select the RNA-seq reads from the tumor sample.*

3. In the next step you will be asked to select the RNA-seq reads from the normal sample (see figure 18.16). Click **Next**.

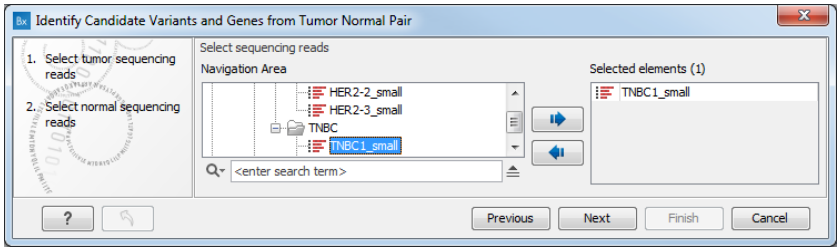


Figure 18.16: *Select the RNA-seq reads from the normal sample.*

4. Configure the parameters for the RNA-Seq Analysis (figure 18.17), first for the tumor sample, and then for the normal sample in the following step.

   If you wish to use spike-in controls, add the relevant file in the "Spike-in controls" field.

   You can also specify that the reads should be mapped only in their forward or reverse orientation (it is by default set to both). Choosing to restrict mapping to one direction is typically appropriate when a strand specific protocol for read generation has been used, as it allows assignment of the reads to the right gene in cases where overlapping genes are located on different strands. Also, applying the 'strand specific' 'reverse' option in an RNA-seq run could allow the user to assess the degree of antisense transcription. Note that mate pairs are not supported when choosing the forward only or reverse only option.

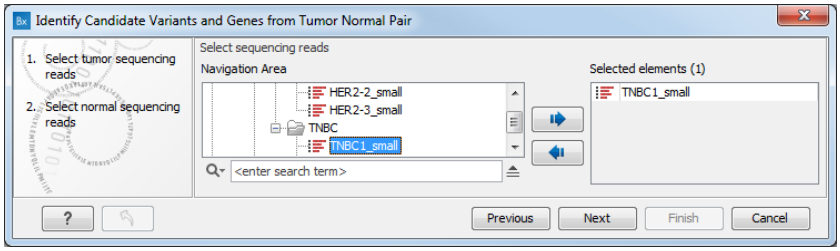


Figure 18.17: *Configure the RNA-Seq Analysis. Here we specified a file for spike-in control but left the strand specific parameter to its default value.*

Click **Next**.

5. Specify in the next 2 windows a **target region** for the analysis of the sample with the **Indels and Structural Variants** tool, first for the **tumor** sample, followed by the **normal** sample (figure 18.18).

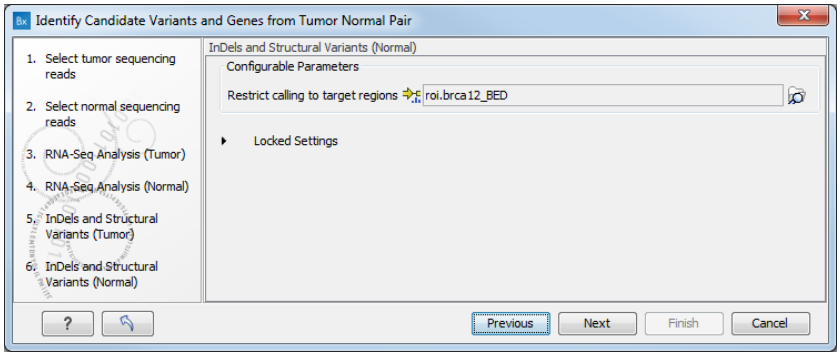


Figure 18.18: *Specify the target region for the Indels and Structural Variants tool.*

The targeted region file is a file that specifies which regions have been sequenced. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents. Remember that you have a hg38-specific BED file when using hg38 as reference, and hg19-specific BED file when using hg19 as reference.

6. In the next wizard step (figure 18.19) you can adjust the settings for the **Create fold change track** tool. This tool calculates for each transcript or gene the ratio between the expression values in the normal and the tumor sample. It becomes then possible to filter on fold changes and expression values, which makes it easy to identify differentially expressed transcripts or genes. The parameters that can be adjusted in this wizard step are described in section 29.3. .

7. Set the parameters for the **Low Frequency Variant Detection** step (see figure 18.20). For a description of the different parameters that can be adjusted in the variant detection step, see section 22.14.

8. The next wizard step (figure 18.21) concerns **removal of germline variants**. You are asked to supply the number of reads in the control data set that should support the variant allele in order to include it as a match. All the variants where at least this number of control reads show the particular allele will be filtered away in the result track.

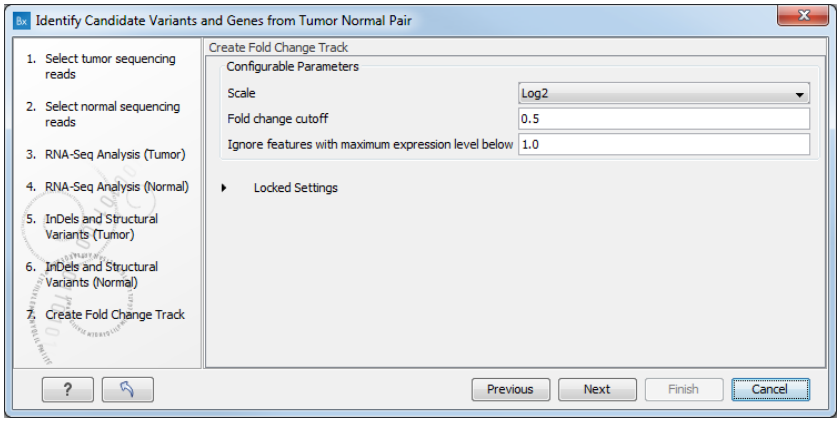


Figure 18.19: *Specify the parameters for variant calling.*

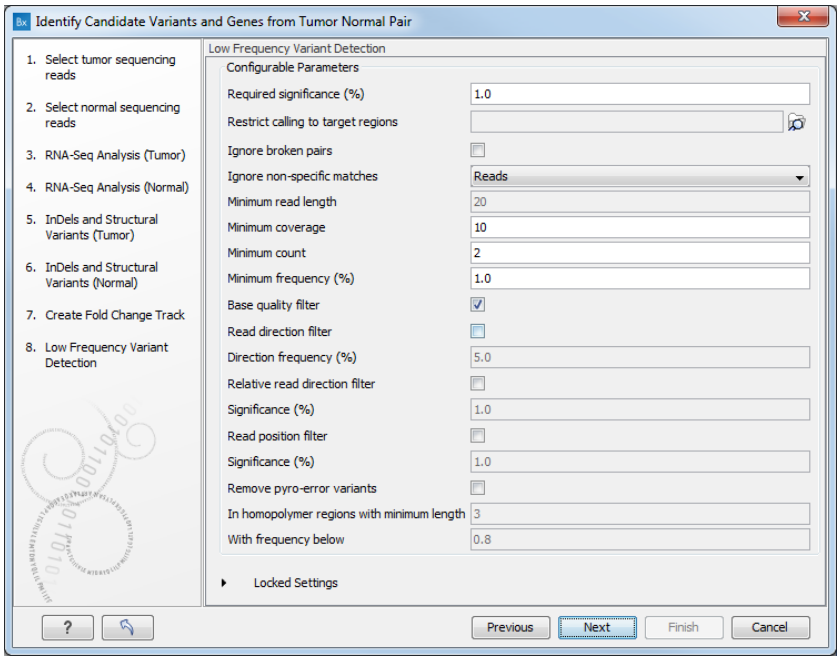


Figure 18.20: *Specify the parameters for variant calling.*

9. Finally, for the Remove Variants Found in HapMap, you can also specify which specific Hapmap population(s) characterize(s) best the samples.

10. Click **Next** to go to the last wizard step. **Preview All Parameters** allows you to view all parameters but not to edit them. Choose to save the results and click **Finish**.

Thirteen types of output are generated:

1. **Gene Expression Normal** and **Gene Expression Tumor** A track showing gene expression annotations. Hold the mouse over or right-clicking on the track. A tooltip will appear with information about e.g. gene name and gene expression values.

2. **Transcript Expression Normal** and **Transcript Expression Tumor** A track showing transcript expression annotations. Hold the mouse over or right-clicking on the track. A tooltip will appear with information about e.g. gene name and transcript expression values.

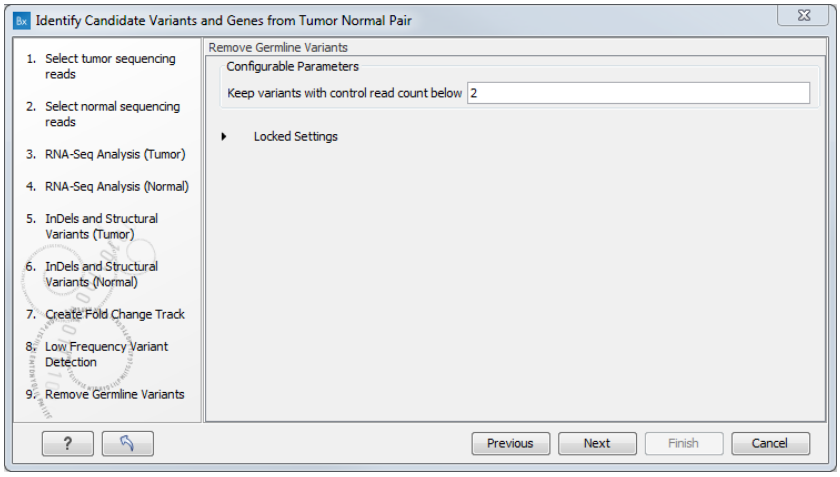


Figure 18.21: *Specify the number of reads to use as cutoff for removal of germline variants.*

3. **RNA-Seq Mapping Report Normal** and **RNA-Seq Mapping Report Tumor** ( ) This report contains information about the reads, reference, transcripts, and statistics. This is explained in more detail in the *Biomedical Genomics Workbench* reference manual in section **RNA-Seq report** (http://resources.qiagenbioinformatics.com/manuals/biomedicalgenomicsworkbench/current/index.php?manual=RNA_Seq_report.html).

4. **Read Mapping Normal** and **Read Mapping Tumor** ( ) The mapped RNA-seq reads. The RNA-seq reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously. For the color codes please see the description in (see section **??**).

5. **Differentially Expressed Genes** file ( ) A track showing the differentially expressed genes. The table view provides information about fold change, difference in expression, the maximum expression (observed in either the case or the control), the expression in the case, and the expression in the control.

6. **Variant Calling Report Tumor** ( ) Report showing error rates for quality categories, quality of examined sites, and estimated frequencies of actual to called bases for different quality score ranges.

7. **Annotated Somatic Variants with Expression Values** ( ) A variant track showing the somatic variants. When mousing over a variant, a tooltip will appear with information about the variant.

8. **Amino Acid Track**

9. **Genome Browser View RNA-Seq Tumor_Normal Comparison** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar and dbSNP (see figure 18.22).

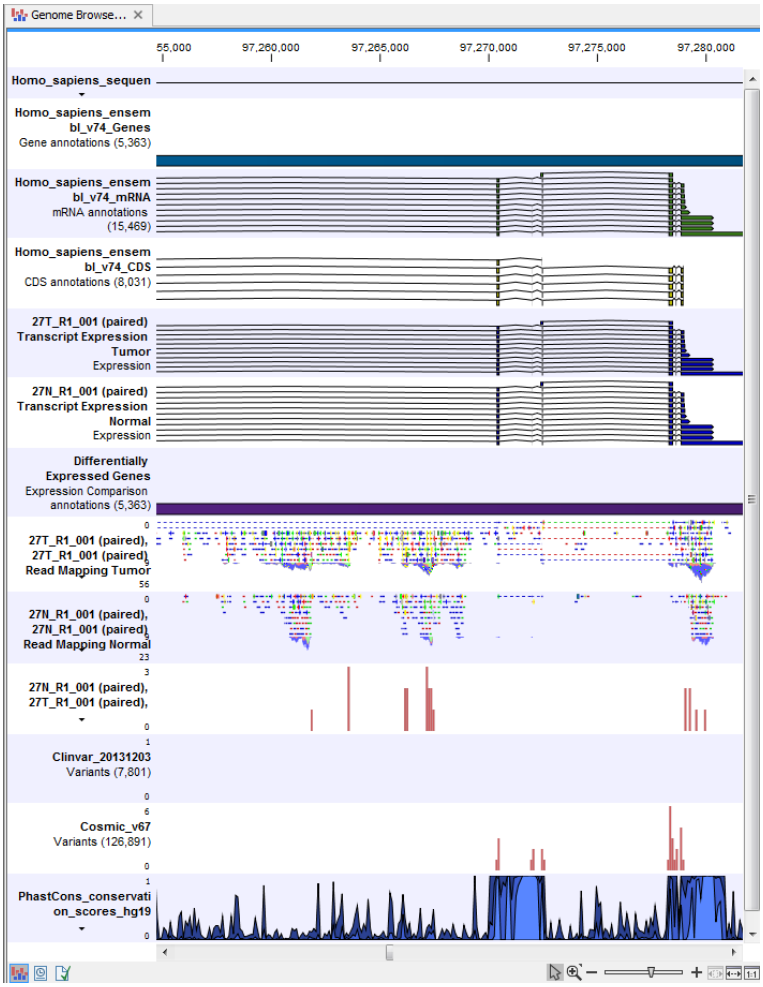


Figure 18.22: *The Genome Browser View is a collection of a number of tracks. The Genome Browser View makes it easy to compare the different tracks. Each track kan be opened individually by double-clicking on the track name in the left side of the View Area.*

18.5 Identify variants and add expression values

The **Identify Variants and Add Expression Values** ready-to-use workflows can be used to identify novel and known mutations in RNA-seq data, automatically map, quantify, and annotate the transcriptomes, and compare the mutational patterns in the samples with the expression values of the corresponding transcripts and genes.

To run the ready-to-use workflow:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Transcriptome Sequencing** ( ) | (**Human**, **Mouse** or **Rat**) | **Identify Variants and Add Expression Values** ( )

1. Double-click on the **Identify Variants and Add Expression Values** tool to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Specify the **RNA-seq reads** to analyze. The reads can be selected by double-clicking on the reads file name or clicking once on the file and then clicking on the arrow pointing to the

right side in the middle of the wizard (figure 18.23). Click **Next**.

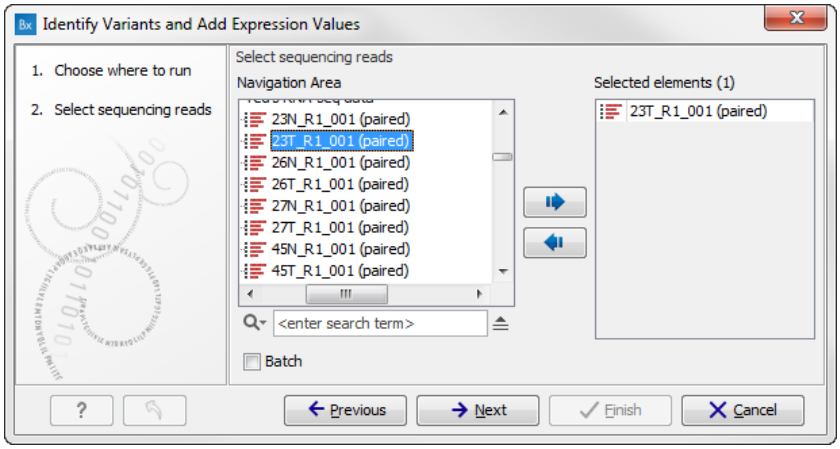


Figure 18.23: *Select the sequencing reads to analyze.*

3. Configure the parameters for the RNA-Seq Analysis (figure 18.24).

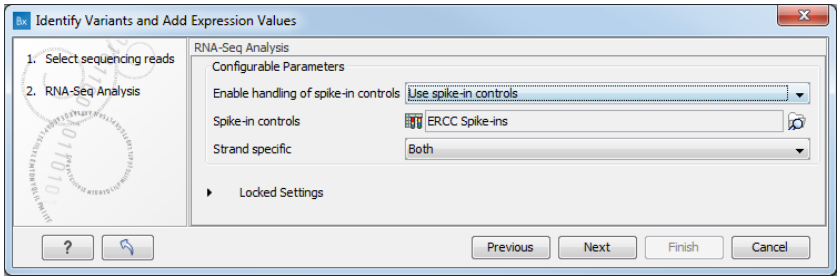


Figure 18.24: *Configure the RNA-Seq Analysis. Here we specified a file for spike-in control but left the strand specific parameter to its default value.*

If you wish to use spike-in controls, add the relevant file in the "Spike-in controls" field.

You can also specify that the reads should be mapped only in their forward or reverse orientation (it is by default set to both). Choosing to restrict mapping to one direction is typically appropriate when a strand specific protocol for read generation has been used, as it allows assignment of the reads to the right gene in cases where overlapping genes are located on different strands. Also, applying the 'strand specific' 'reverse' option in an RNA-seq run could allow the user to assess the degree of antisense transcription. Note that mate pairs are not supported when choosing the forward only or reverse only option.

Click Next.

4. Specify a **target region** for the **Indels and Structural Variants** tool (figure 18.25).

   The targeted region file is a file that specifies which regions have been sequenced. This file is something that you must provide yourself, as this file depends on the technology used for sequencing. You can obtain the targeted regions file from the vendor of your targeted sequencing reagents. Remember that you have a hg38-specific BED file when using hg38 as reference, and hg19-specific BED file when using hg19 as reference.

5. Set the parameters for the **Low Frequency Variant Detection** step (see figure 18.26). For a description of the different parameters that can be adjusted in the variant detection step,

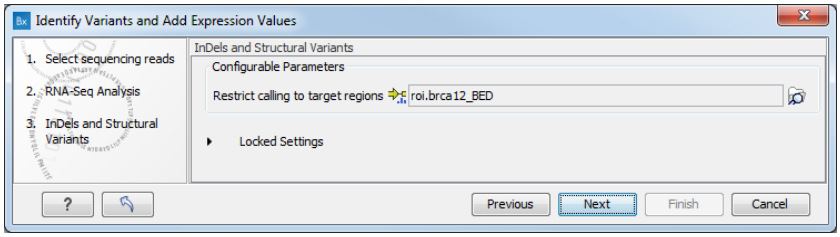


Figure 18.25: *Specify the target region for the Indels and Structural Variants tool.*

see section 22.14.

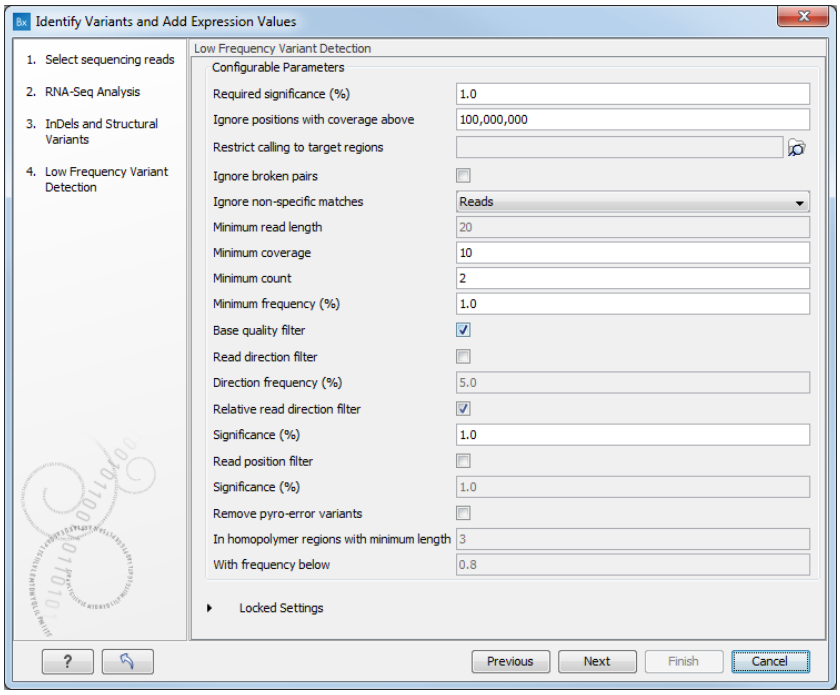


Figure 18.26: *Specify the parametes for transcriptomic variant detection.*

6. If you are working with the workflow from the Human folder, specify here the relevant **1000 Genomes** population (and **HapMap** populations at the next step) from the drop-down list (see figure 18.27). Choose the population that matches best the population your samples are derived from.

   Under "Locked settings" you can see that "Automatically join adjacent MNVs and SNVs" has been selected. The reason for this is that many databases do not report a succession of SNVs as one MNV as is the case for the *Biomedical Genomics Workbench*, and as a consequence it is not possible to directly compare variants called with *Biomedical Genomics Workbench* with these databases. In order to support filtering against these databases anyway, the option to **Automatically join adjacent MNVs and SNVs** is enabled. This means that an MNV in the experimental data will get an exact match, if a set of SNVs and MNVs in the database can be combined to provide the same allele.

   Note: This assumes that SNVs and MNVs in the track of known variants represent the same allele, although there is no evidence for this in the track of known variants.

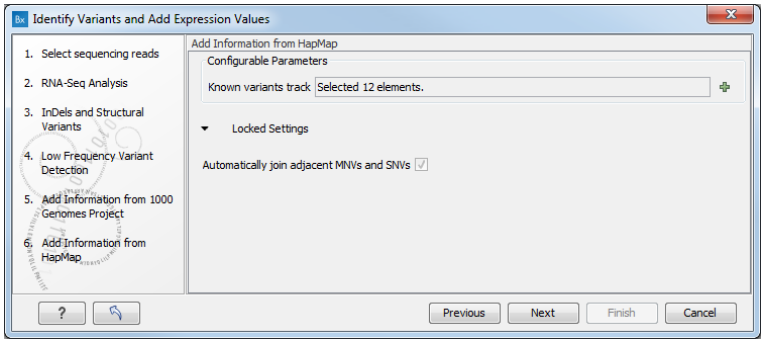


Figure 18.27: *Select the relevant population from the drop-down list for 1000 Genomes and Hapmap databases.*

7. Click **Next** to go to the last wizard step. **Preview All Parameters** allows you to preview all parameters but not edit them. Choose to save the results and click **Finish**.

Seven different output types are generated:

1. **Gene expression** ( ) A track showing gene expression annotations. Hold the mouse over or right-clicking on the track. A tooltip will appear with information about e.g. gene name and expression values.
2. **Transcript expression** ( ) A track showing transcript expression annotations. Hold the mouse over or right-clicking on the track. A tooltip will appear with information about e.g. gene name and expression values.
3. **RNA-Seq Mapping Report** ( ) This report contains information about the reads, reference, transcripts, and statistics. This is explained in more details here `http://resources.qiagenbioinformatics.com/manuals/biomedicalgenomicsworkbench/current/index.php?manual=RNA_Seq_report.html`.
4. **Read Mapping** ( ) The mapped RNA-seq reads. The RNA-seq reads are shown in different colors depending on their orientation, whether they are single reads or paired reads, and whether they map unambiguously. For the color codes please see section 22.2.2.
5. **Annotated Variants with Expression Values** ( ) Annotation track showing the variants. Hold the mouse over one of the variants or right-clicking on the variant. A tooltip will appear with detailed information about the variant.
6. **RNA-Seq Genome Browser View** ( ) A collection of tracks presented together. Shows the annotated variants track together with the human reference sequence, genes, transcripts, coding regions, and variants detected in ClinVar and dbSNP (see figure 18.14).
7. **Log** ( ) A log of the workflow execution.

18.6 Identify and Annotate Differentially Expressed Genes and Pathways

The **Identify and Annotate Differentially Expressed Genes and Pathways** compares genes expression in different groups of samples and performs a gene ontology (GO) enrichment

analysis on the differentially expressed genes to identify affected pathways. The workflow takes as input Gene Expression (GE) or Transcript Expression (TE) tracks that were generated using the RNA-Seq analysis tool. The samples must be associated to a metadata table.

To run the ready-to-use workflow:

> **Toolbox** | **Ready-to-Use Workflows** | **Whole Transcriptome Sequencing** ( ) | **Human**, **Mouse** or **Rat** | **Identify and Annotate Differentially Expressed Genes and Pathways** ( )

1. Double-click on the **Identify and Annotate Differentially Expressed Genes and Pathways** ready-to-use workflow to start the analysis. If you are connected to a server, you will first be asked where you would like to run the analysis.

2. Next, you will be asked to select the samples to analyze (figure 18.28). You can select several GE tracks or TE tracks generated by the RNA-Seq analysis tool, but not a combination of both. Click **Next**.

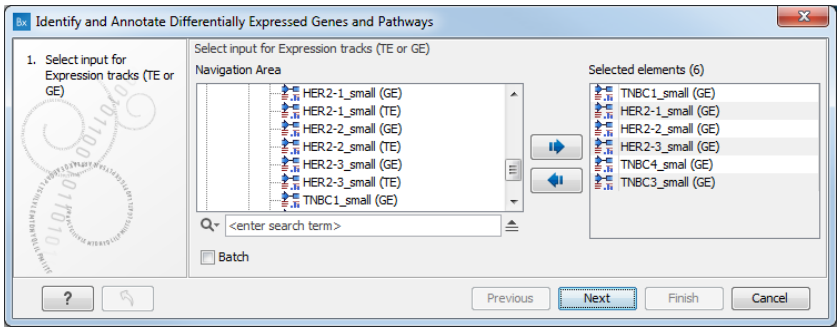


Figure 18.28: *Select the GE or TE tracks to analyze.*

3. In the next wizard step you can set up the experimental design associated with the data (figure 18.29):

   - Choose the metadata table that was associated to the GE or TE tracks used in the previous step.
   - Choose the factor (must be one of the metadata category) that should be used to test for differential expression.
   - It is possible to specify confounding factors, i.e., factors that are not of primary interest, but may affect gene expression.
   - The Comparisons panel determines the number and type of statistical comparison tracks output by the workflow (see section 29.5.2 for more details).

4. In the next step you can choose to preview the settings and save the results (see figure 18.30).

   Click **Finish** to start the analysis.

The following outputs are generated:

1. **PCA for RNA-Seq** plot ( ) Projects a high-dimensional dataset (where the number of dimensions equals the number of genes or transcripts) onto two or three dimensions.

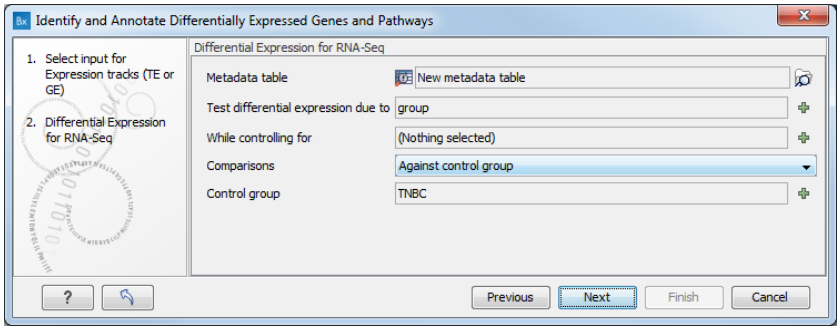


Figure 18.29: *Specify the experimental design desired for running the workflow.*

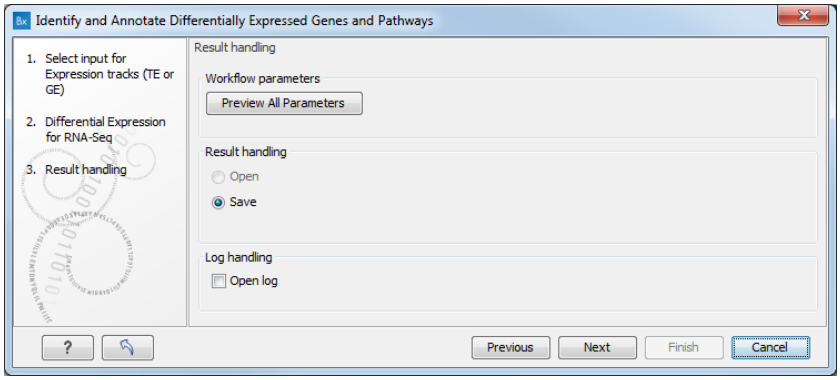


Figure 18.30: *The results handling step.*

2. **Statistical Comparison** ( ) The information can be accessed in two different ways:

   - Open as a track, hold shift and hover over a feature. A tooltip will appear with information about gene name, results of statistical tests, and expression values.
   - Open the track in table format by clicking on the table icon in the lower left side of the View Area.

3. **Genome Browser View Differentially Expressed Genes and Pathways** ( ) A collection of tracks presented together. Shows the human reference sequence, annotation tracks for genes, coding regions CDS, mRNA, and statistical comparison tracks (see figure 18.31).

4. **Heat Map for RNA-Seq** ( ) A two dimensional heat map of expression values. Each column corresponds to one sample, and each row corresponds to a feature (a gene or a transcript). The samples and features are both hierarchically clustered.

5. **Venn Diagram** ( ) To compare the overlap of differentially expressed genes or transcripts in two or more statistical comparison tracks.

6. **Expression Browser** ( ) To inspect gene and transcript expression level counts and statistics for many samples at the same time.

7. **GO Enrichment Analysis** ( ) A table showing the results of the GO enrichment analysis. The table includes GO terms, a description of the affected function/pathway, the number of genes in each function/pathway, the number of affected genes within the function/pathway, and p-values.

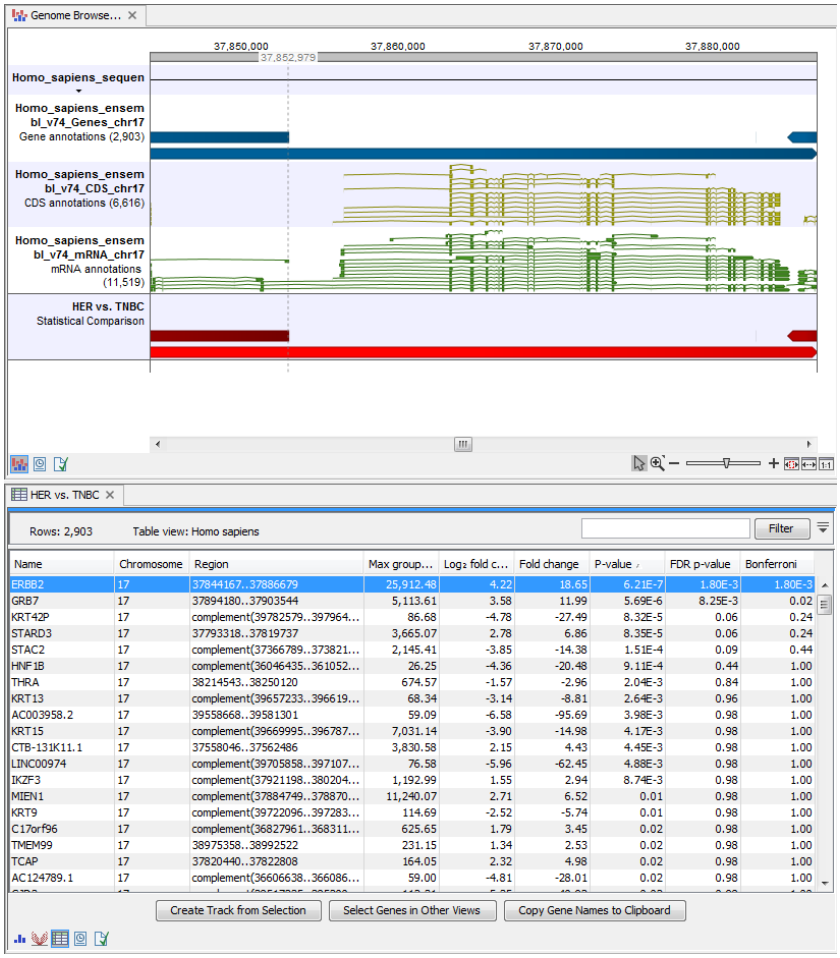


Figure 18.31: *The genome browser view allows comparison of the expression comparison tracks with the reference sequence and different annotation tracks.*

Please refer to the relevant sections of the chapter 29 for additional information on the different output mentioned above.

Part IV

CLC Genome Browser

Chapter 19

Genome browser

Contents

This chapter explains how to visualize tracks, how to retrieve reference data and finally how to perform generic comparisons between tracks.

The genome browser is the graphical interface where tracks can be presented alone or together with other tracks. Tracks are the fundamental building blocks for data analysis in the *Biomedical Genomics Workbench* and provide a unified framework for the visualization, comparison and analysis of genome-scale studies.

In tracks, all information is tied to genomic positions. A central coordinate-system is provided by a reference genome, which allows different types of data or results for different samples to be seen and analyzed together.

Different types of data are represented in different types of tracks, and each type of track has its own particular editors. An example of a paired-end mapping read-track displaying reads and coverage is shown in figure 19.1.

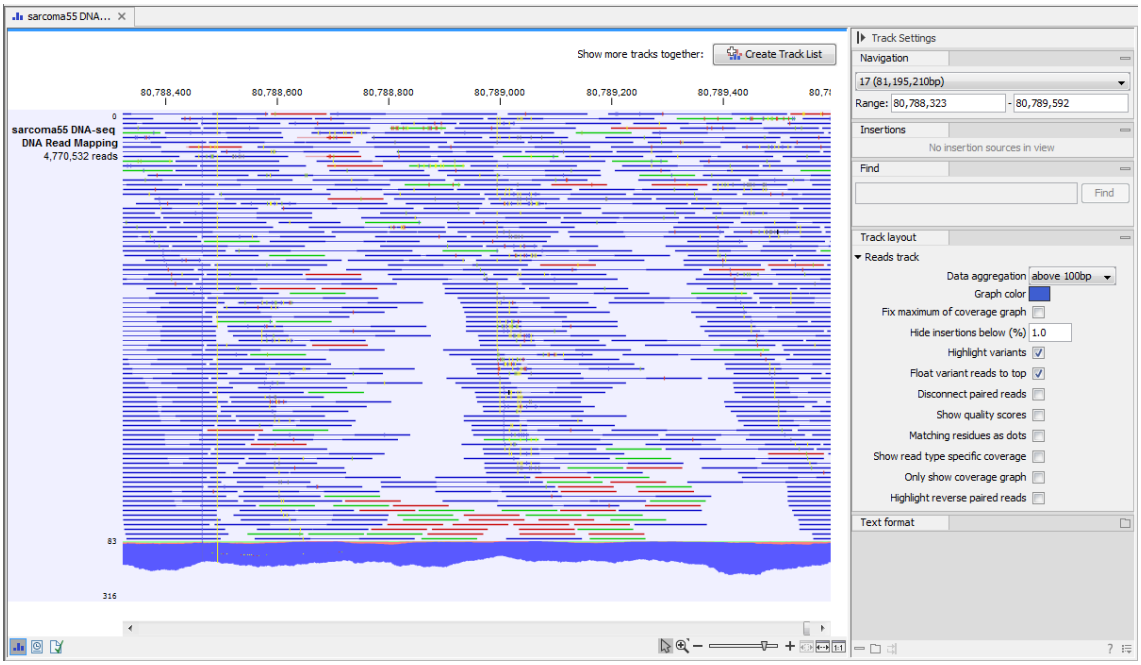


Figure 19.1: *A paired-end mapping read-track opened, displaying reads and coverage. On the top right, the button for creating a Track List is visible. On the right is the Side Panel.*

19.1 Track types

The different track types in the *Biomedical Genomics Workbench* are:

A sequence ( ) This is the track type that is used for holding the reference genome. The sequence track contains the single reference sequences of the genome (e.g. the chromosomes or the consensus sequences of de novo assembled contigs).

A reads track ( ) This is the track type that is used for holding a read mapping e.g. as produced by the Map Reads to Reference (see section 22.1) or Local Realignment (see section 22.5) tools. The reads track contains all the reads that have been mapped at their mapped positions, and you can zoom in all the way to base resolution. In case there are more reads than the height of the track allows, an overflow graph will be displayed below the reads in the same colors than the reads that it represents.

A variant track ( ) A variant track is a particular kind of track that is used to store features that fulfill the requirements for being a variant. A particular requirement for being a variant is that it refers to a particular region of the reference, and it is possible to describe exactly how the sample "Allele" sequence looks in this region, as compared to what the "reference allele" sequence looks like in this region. Variants may be of type SNV, MNV, replacement, insertion or deletion. A variant track may be produced either by running a Variant detection analysis (e.g using a variant caller or by importing a variant format file (such as a "vcf" or a "gvf" file) or downloading it from a database (e.g. dbSNP). The tool InDels and Structural Variants (see section 22.9) detects structural variants, including insertions, deletions, inversions, translocations and tandem duplications. It will produce a variant track, which will contain some insertions and deletions (the "InDel" track). However, the tool will also detect some insertions for which the "Allele" sequence is not fully, but only partially, known.

These insertions do not fulfill the requirements of being a variant and therefore cannot be put in the variant track. Instead they are put in the "SV track", along with the inversions and translocations. The "SV" track is an "annotation" (or "feature") track, which is less strict and more flexible, in the requirements to the types of annotations (or features) that it can contain (see below).

An annotation track ( ) Each annotation track contains a certain type of annotations. Examples are gene or mRNA tracks, which contain gene, respectively mRNA, annotations, UTR tracks, conservation score tracks and target region tracks. They may be obtained either by importing (see section 6.2 or downloading them into the Workbench (e.g from a .bed, .gtf or .gff file or a database, such as ENSEMBL). Also, many of the tools in the *Biomedical Genomics Workbench* will output annotation tracks. Examples are the Indels and Structural Variants tool, which will put the detected structural variants (that do not fulfill the requirements for being of type "variant") in an annotation track, or the ChIP-Seq detection tool which will put the detected "peaks" into a "peak" annotation track.

A coverage graph ( ) The coverage graph track is calculated from a reds track and contains a graphical display of the coverage at each position in the reference.

An expression track ( ) The RNA-seq algorithm produces expression tracks: one for genes and one for transcripts. These have an annotation for each gene or transcript, and an expression value associated to that annotation. The type of expression value associated with each annotation is determined by the expression value parameter selected in the RNA-Seq tool. These values are visualized as a color gradient from blue to red; the lowest expression value within each chromosome of the track is represented as 0% and the highest expression value within each chromosome of the track is represented by 100%.

An example of the different types of tracks is given in figure 19.2.

19.1.1 Visualizing, zooming and navigating tracks

The Side Panel is shown to the right of a track opened in the View Area. The settings available in the side panel are specific to the type of track open, but usually allow users to navigate the track using a specific position on a specific chromosome, to find a particular nucleotide sequence or annotation, and to change the text format. In addition, it is possible to change the track layout. For example, you can learn more about Reads track side panels here: section 22.2.3. Once you have changed the visualization of the track to your liking, it is possible the settings as described in section 4.6.

It is possible to zoom in and out in a track using the buttons in the lower right-hand corner of the View Area.

- to zoom in to 100 % to see the data at base level, click the **Zoom to base level** ( ) icon.
- to zoom out to see all the data, click the **Zoom to Fit** ( ) icon.

When zooming out you will see that the data is visualized in an aggregated format using a density bar plot or a graph.

You can also use the zoom and scroll shortcuts described in the table below:

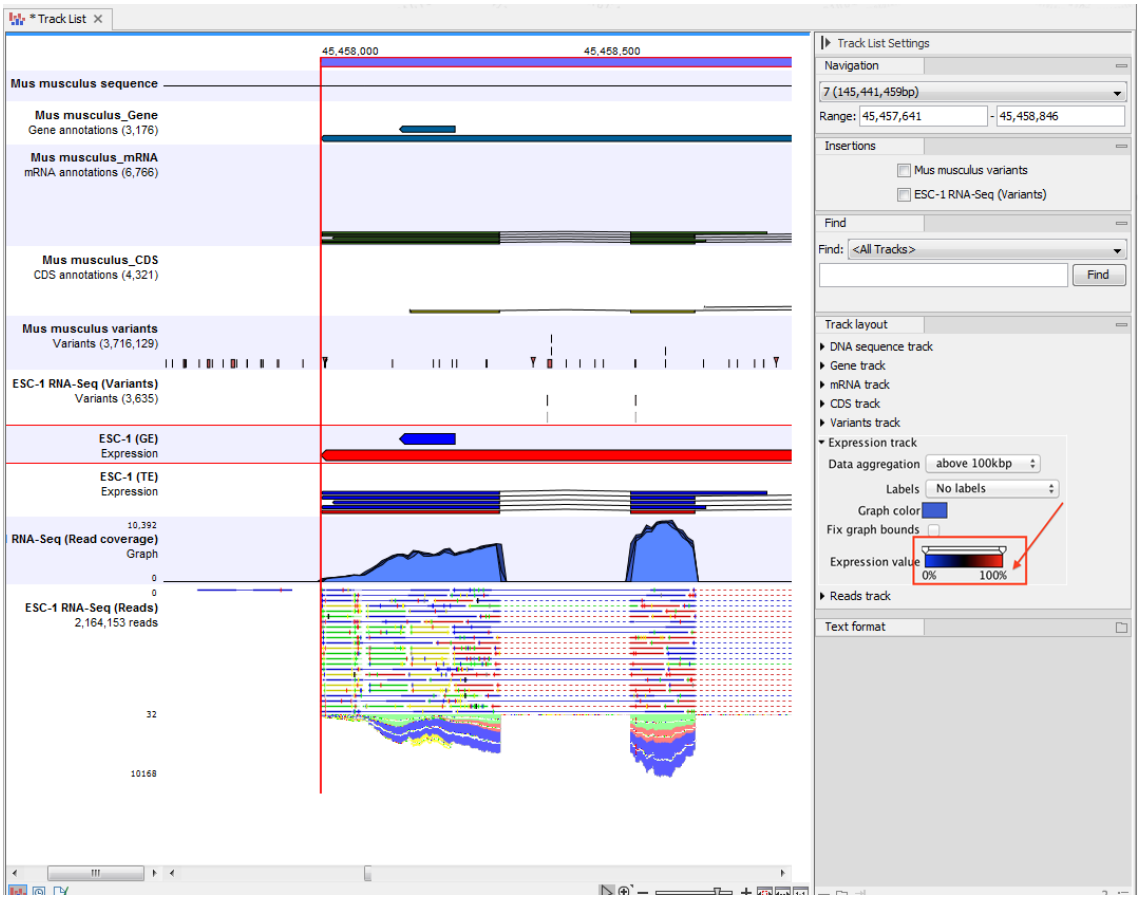


Figure 19.2: *A tracklist containing different types of tracks. From the top: a sequence track, three annotation tracks with gene, mRNA and CDS annotations respectively, two variant tracks, a gene-level (GE) and a transcript level (TE) expression track, a coverage track and a reads track.*

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

It is possible to open some tracks as a table. When that is the case, it is usually helpful to open the table in split view so that both the track and the table can be seen at the same time. In particular, because track and table are connected, it becomes possible to navigate and zoom the track by selecting successively the different rows in the table. To open a track as a table in split view, press Ctrl (⌘ on Mac) while you click the table button at the bottom of the track view.

You can also right-click on the track tab, and select "Show View | Table".

19.2 Create new genome browser view

The tool **Create New Genome Browser View** ( ) can be used to create a list of tracks. Double-click on **Create New Genome Browser View** in the toolbox to run the tool:

Toolbox | **Genome Browser** ( ) | **Create New Genome Browser View** ( )

In the wizard (figure 19.3) you can select all the tracks that you would like to include in your Genome browser view. Figure 19.4 shows an example of a genome browser view including a track with the genomic reference sequence at the top followed by the targeted regions, the mapped reads, and in the lower part of the figure a variant detection track.

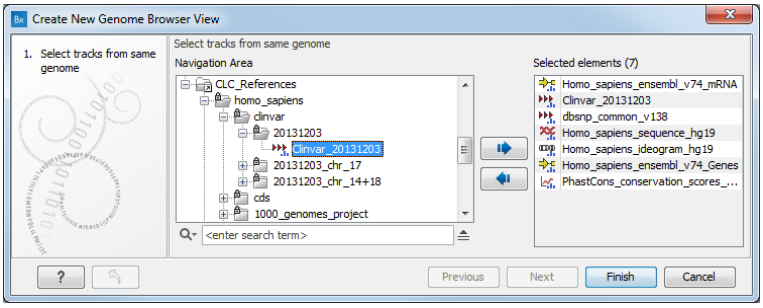


Figure 19.3: *Select all the tracks you would like to include in your Genome browser view.*

Figure 19.4: *Seven tracks shown in the Genome Browser View.*

19.3 Genome browser view tools

For details on how to find and import different tracks see section 6.2. Tracks are saved as files in the **Navigation Area** with specific icons representing each track type, e.g. an annotation track ( ).

To visualize several tracks together, they can be combined into a **Genome Browser View** ( ). Genome browser views can be created in different ways. One way is via the menu bar:

File | **New** | **Create Genome Browser View ( )**

The track list is designed to be used as a container for multiple tracks for easy visualization and comparative analysis. Therefore all the involved tracks and the track list are required to be present and located in one single location (Workbench or CLC Server). Otherwise, they will be marked as "Unresolved track" in the track list.

19.3.1 Adding ideogram to Genome Browser View

An ideogram, also called a cytogenetic ideogram, is a chromosome map with numbered banding patterns that shows the relationship between the two chromosome arms and the centromere. Users of the UCSC Genome Browser (`http://genome.ucsc.edu/`) are most likely familiar with the use of cytogenetic ideogram in the UCSC Genome Browser. Ideograms can be imported into the Workbench (from UCSC) and be used in track lists to provide an overview of the analyzed data in context of the individual chromosomes. An example is shown in figure 19.5.

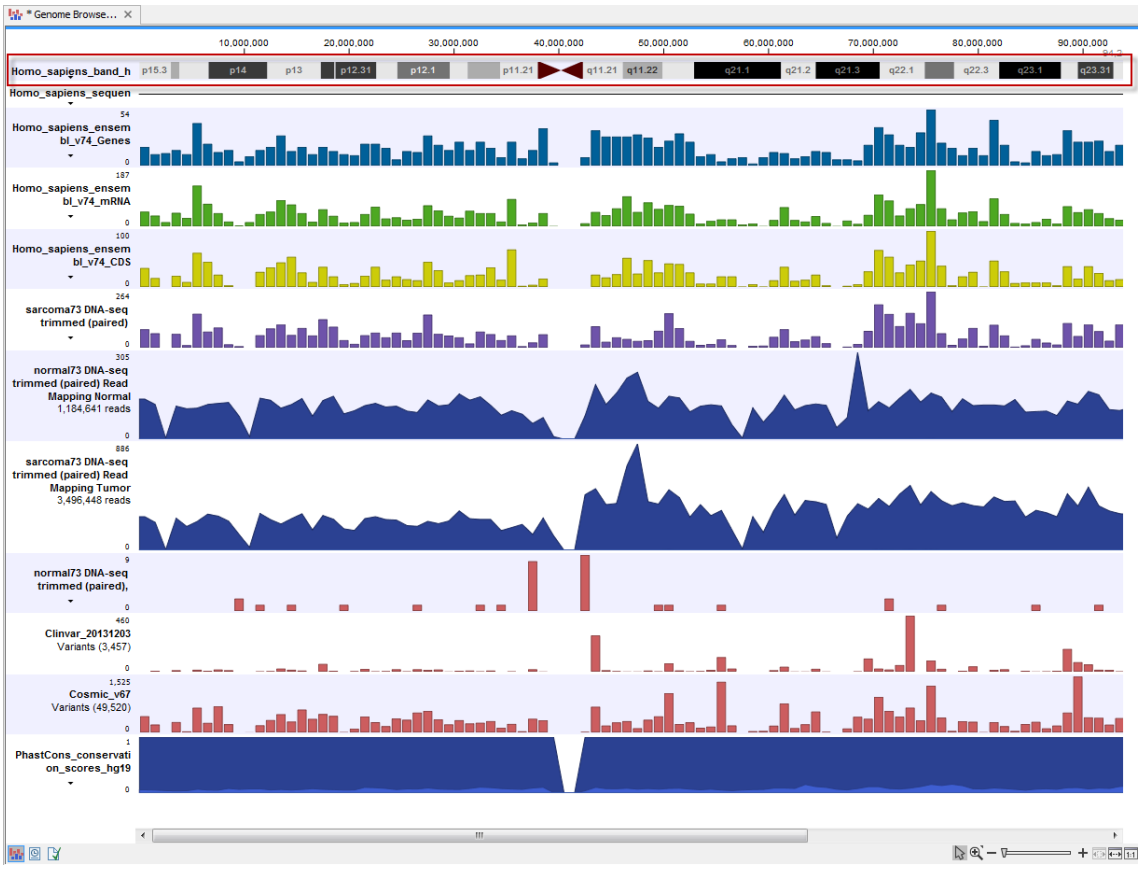


Figure 19.5: *The ideogram is particularly useful when used in combination with other tracks in a track list. In this figure the ideogram is highlighted with a red box.*

The ideogram is automatically imported into *Biomedical Genomics Workbench* along with the reference data the first time the reference data is imported into the *Biomedical Genomics Workbench*. This is done with the Data Management function found in the upper right corner of the Workbench.

Data Management ()

For a description of how to import reference data into *Biomedical Genomics Workbench* we refer to section 13.1.

The ideogram can be found in the **Navigation Area** in the **CLC_References** folder as shown in figure 19.6. To include an ideogram in a Genome Browser View, you can click once on the ideogram in the **Navigation Area** and while holding down the mouse key, you can drag the ideogram into an Genome Browser View that is open in the **View Area**.

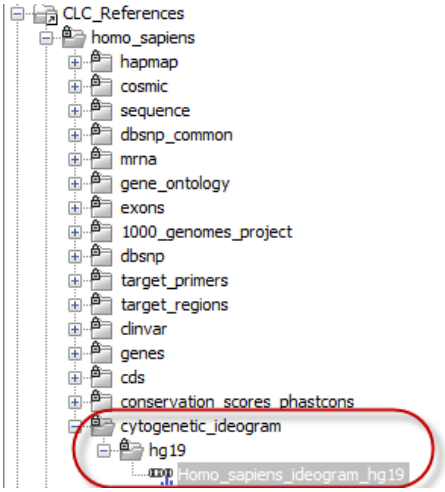


Figure 19.6: *The ideogram is part of the reference data and is particularly useful when used in combination with other tracks in a Genome Browser View.*

19.3.2 Adding, removing and reordering tracks

You can organize your tracks by dragging them up and down. Right-clicking on any of the tracks opens up a context menu with several options (Figure 19.7). The options shown in the context menu will vary depending on which tracks you have open in the viewing area. Hence, you may not be presented with all the options described here.

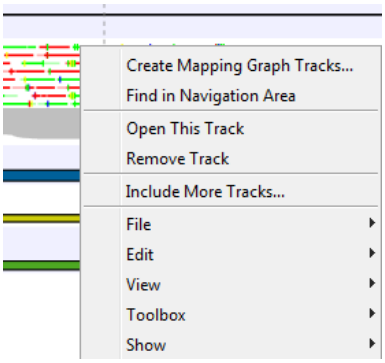


Figure 19.7: *Options to handle and organize tracks.*

Create Mapping Graph Tracks This will allow you to create a new track from a mapping track (learn more in section 19.4).

Find in Navigation Area This will select the track in the **Navigation Area**.

Open This Track This opens a new view of the track. For annotation and variant tracks, a table view is opened as described in section 19.3.3. This can also be accomplished by double-clicking the track.

Remove Track This will remove the track from the current view. You can add it again by dragging it from the **Navigation Area** into the track list view or by pressing **Undo** ( ).

Include More Tracks This will allow you to add other track sets to your current track set. Please note that the information in the track will still be stored in its original track set. This means that you by including a track in this way at the same time is adding a reference to this track in another track set. An example of this could be the inclusion of a SNP track from another sample to your current analysis.

19.3.3 Showing a track in a table

All tracks containing annotations (including variants) can be opened in a table.

From the track list (see section 19.3) this is done either by double-clicking the label of the track or by right-clicking the track and choosing **Open This Track**. Alternatively, you can open the track from the **Navigation Area** and switch to the table view ( ) at the bottom.

The table will have one row for each annotation, and the columns will reflect its information content. Figure 19.8 shows an example of a variant database track that is presented in a table.

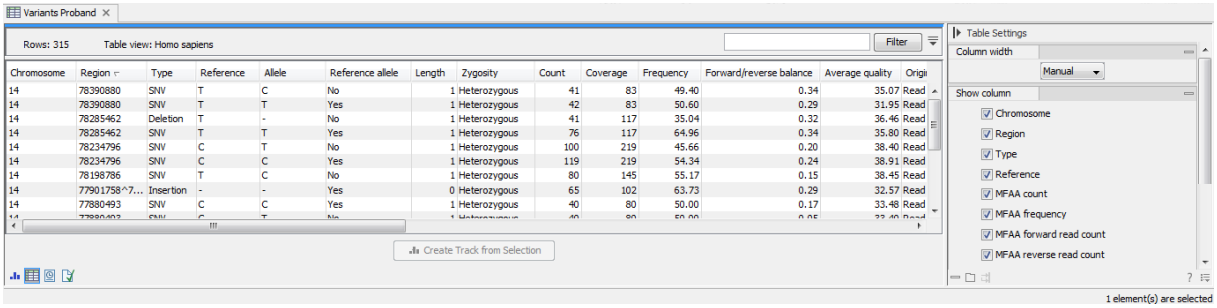


Figure 19.8: *Showing a variant track in a table view.*

You can use the table to sort, filter and select annotations (see Appendix 3.3). Please note that there are two additional options for *filtering on overlaps* in the "Region" column

When selecting a row in the table the graphical view will jump to this position on the genome. Please note that table filtering only affects the table. The track itself remains unaffected and keeps all annotations. If you also wish to filter tracks in the graphical view, the **Annotate and Filter** tools can be used instead.

At the bottom of the table a button labeled **Create Track from Selection** is available. This function can be used to create tracks showing only a subset of the data and annotations. Select the relevant rows in the table and click the button to create a new track that only includes the

selected subset of the annotations. This function is particularly useful when used in combination with the filter.

19.3.4 Open track from a track list in table view

To open a table view of a track that is part of a track list, open the track list by double-clicking on the track name in the **Navigation Area**. The track will open in a graphical view. To open a single track from the track list in table view, either right-click on the track and choose "Open This Track" (see figure 19.9) or double-click on the name of the track you would like to open in table view (in the left side of the track when it is open in the **View Area**. This will automatically open op the specific track in table view.

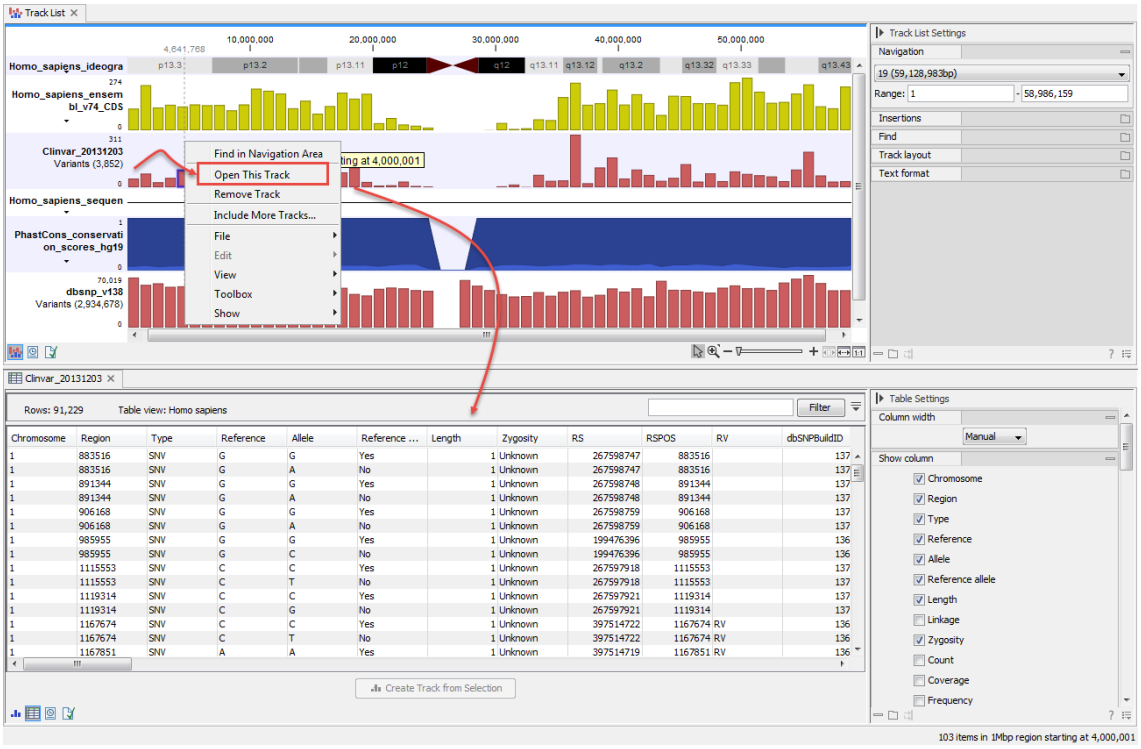


Figure 19.9: *One way to open a table view of a track that is part of a track list is to right click on the track of interest and select "Open This Table".*

19.3.5 Finding annotations on the genome

In the **Side Panel** under **Find**, a search field allows you to quickly find the annotation that you are looking for. The list of tracks further allows you to restrict the search to a particular track (e.g. a gene track).

In the search field you can enter any kind of text that exists in the annotation track. As an example, consider the gene and tool tip shown in figure 19.10.

This gene could be found by searching for the name specifically or, by using a wildcards (asterisks), less specific search terms could be use, giving more flexibility. To find BRCA2, for example, any of the following entries could be typed in the search field:

BRCA2 This would match the annotation name exactly.

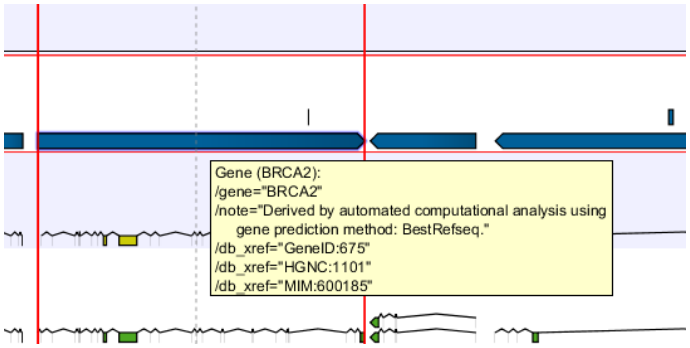


Figure 19.10: *The BRCA2 gene.*

BRCA* This would match the annotation name as well as other genes with a text starting with BRCA (e.g. the BRCA1 gene).

***RCA2** This would match the annotation name as well as other genes with a text ending with RCA2 (e.g. the SMARCA2 gene).

600185 This would match the `db_xref` qualifier for the OMIM database. All the text shown for the annotation in figure 19.10 can be searched this way, both as exact matches and with the * before or after the search term.

Just below the search field in the **Side Panel**, a status label informs about the progress of the search and the hit that has been found. Placing the mouse on top of the label will display a tool tip with more info (see 19.11).

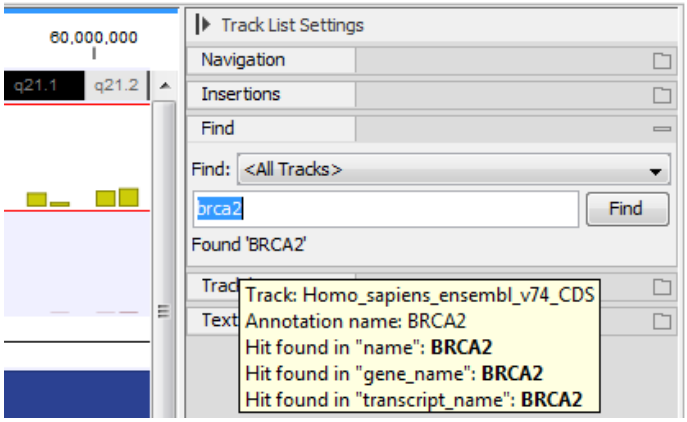


Figure 19.11: *The BRCA2 gene found.*

The search will be performed throughout the entire genome beginning with the chromosome currently shown and stopping when it finds the first hit. Press **Find** again to find the next hit. Once the whole genome has been traversed, the status will inform you that you have searched the whole genome. Click the **Find** button to start the search again.

Please note that you can also use the table view of an annotation track to perform more advanced queries of the data (see section 19.3.3).

19.3.6 Extract sequences from tracks

It is possible to extract sequences from tracks. The sequence of interest can be selected by

dragging the mouse over the region of interest followed by a right click on the reads and a click on **Extract sequences** (figure 19.12).

Figure 19.12: *Extract sequences from tracks.*

This opens up the dialog shown in figure 19.13 that allows specification of whether the selected sequences should be extracted as single sequences or as a list of sequences.

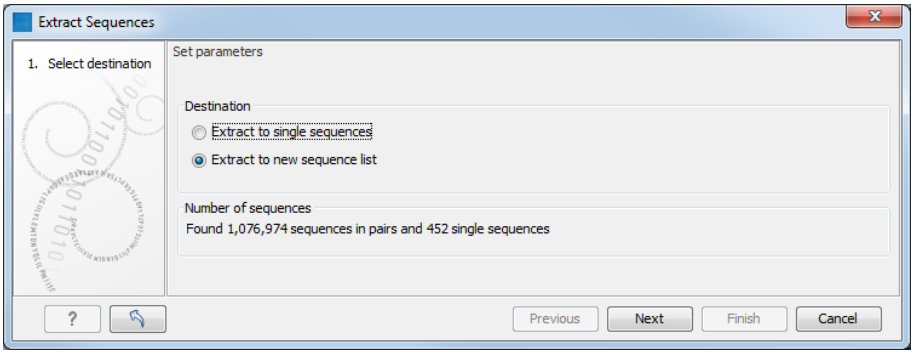


Figure 19.13: *Select destination for extracted sequences.*

Right clicking on the reads also enable the option **Extract from selection**, a function that corresponds to the **Extract from selection** described in section 33.7.5 although with small differences. Common for both versions of the **Extract from selection** function is that when extracting reads in an interval, only reads that are completely covered by the selection will be part of the extracted sequence, which in turn means that the tool can be used to extract only a subset of reads.

Clicking **Extract from selection** opens up the dialog shown in figure 19.14.

The purpose of this dialog is to let you specify which kinds of reads you wish to include. Per default all reads are included.

The options are:

Interval

Only include reads contained within the intervals Only reads that are included within the selection will be extracted. Reads that continue outside the selected area are not included.

Paired status

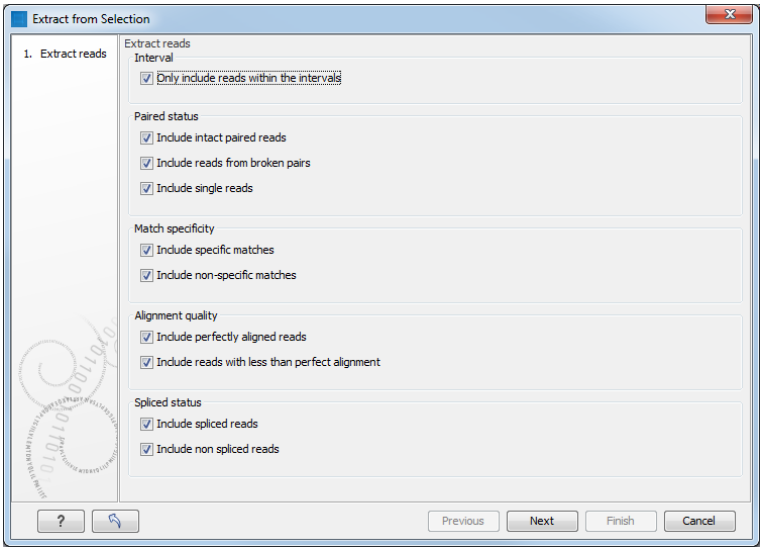


Figure 19.14: *Select the reads to include.*

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.

19.3.7 Creating track lists in workflows

Track lists can be created as part of workflows. Track lists are different from all other workflow outputs in the sense that the tracks inside the track lists have to be saved separately, even if they are included in a track list.

Figure 19.15 shows an example where two tracks are fed into the **Create New Genome Browser View** element.

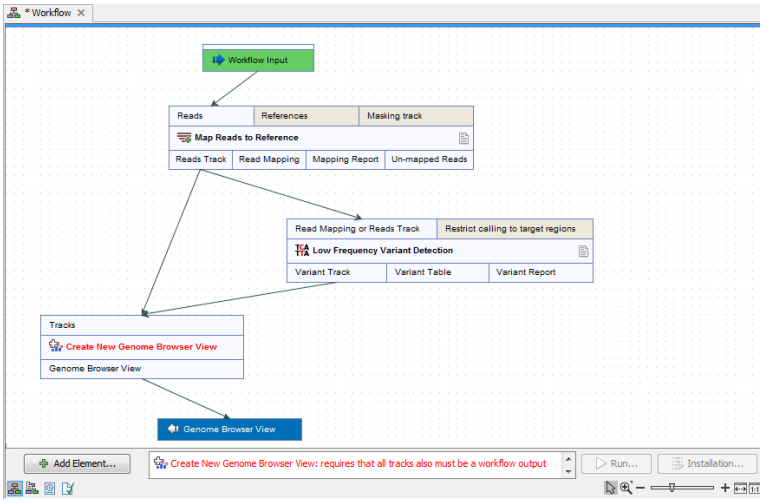


Figure 19.15: *This workflow does not work because the two tracks need to be marked as output.*

In this example, there is a warning at the bottom of the editor pointing at the fact that these two tracks need to be selected as output in order for the workflow to be validated. In figure 19.16, this has been corrected by selecting the tracks as output, and the workflow can now be executed.

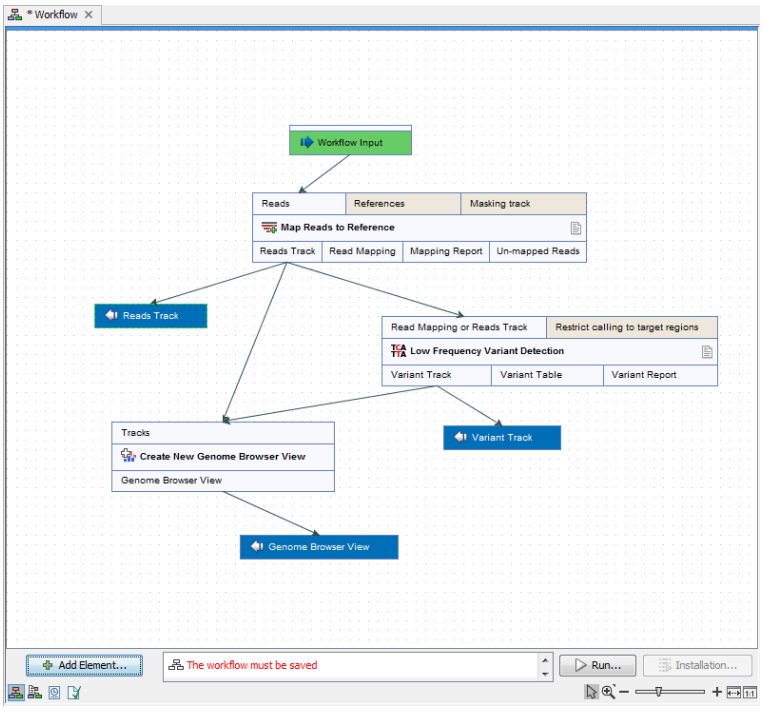


Figure 19.16: *The tracks have been selected as output, and the workflow can now be executed.*

19.4 Graphs

Graphs can be a good way to quickly get an overview of certain types of information. This is the case for e.g. the GC content in a sequence or the read coverage. The *Biomedical Genomics Workbench* offers two different tools that can create graph tracks from either a sequence or a read mapping. The two available tools are:

- **Create GC Content Graph**
- **Create Mapping Graph**

Both tools are found in the toolbox:

Toolbox | **Genome Browser** () | **Graphs**

Graph tracks can also be created directly from the track view or track list view by right-clicking the track you wish to use as input, which will give access to the toolbox.

To understand what graph tracks are, we will look at an example. We will use the **Create GC Content Graph** tool to go into detail with one type of graph tracks.

19.4.1 Create GC Content Graph

The **Create GC Content Graph** tool needs a sequence track as input and will create a graph track with the GC contents of that sequence.

To run the tool go to the toolbox:

Toolbox | **Genome Browser** () | **Graphs** | **Create GC Content Graph**

Select the sequence track that should be used as input (see figure 19.17).

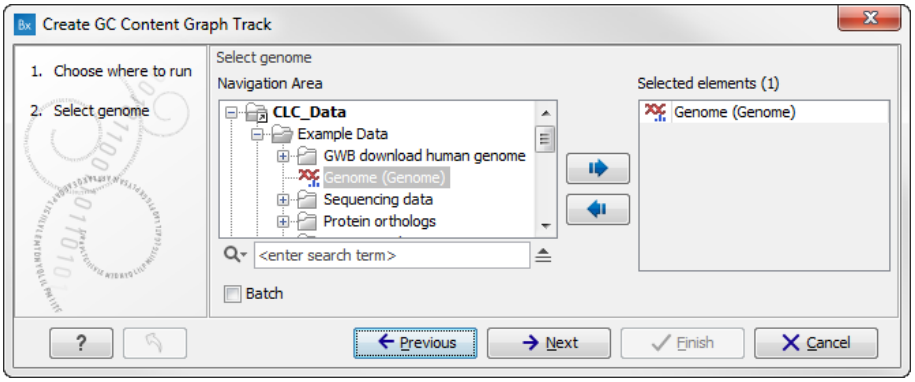


Figure 19.17: *Select the sequence track that should be used as input.*

In the next wizard step (see figure 19.18), you can specify the window size, i.e., the size of the window around the central base in the region that is used to calculate the GC content. This number must be odd as you need a central base and an equal number of bases to each side of the central base. For example, with a window size of 25, the GC content for the central base will be calculated based on the nucleotide composition in the central base and the 12 bases upstream and 12 bases downstream of the central base.

Click on the button labeled **Next**, choose to save your results, and click on the button labeled **Finish**. The output can be seen in figure 19.19. The output from "Create GC Content Graph" is a

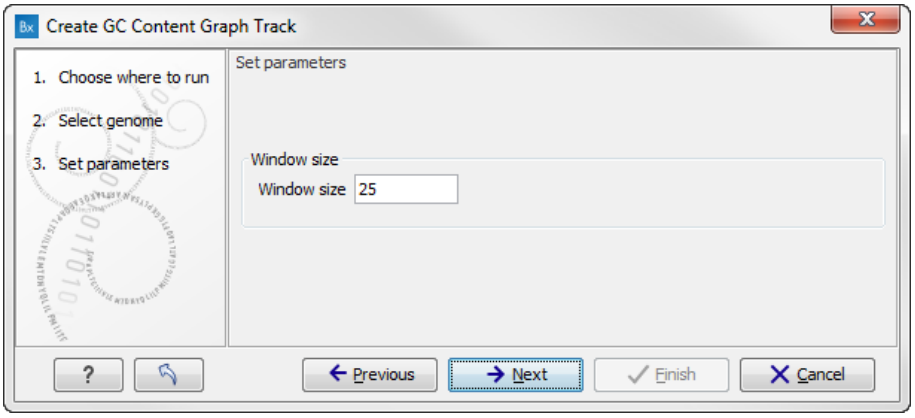


Figure 19.18: *Specify the window size. The window size is the region around each individual base that should be used to calculate the GC content in a given region.*

graph track. The graph track shows one value for each base with one graph being available for each chromosome. When zoomed out as shown in this figure, three different graphs with three different colors can be seen. The top graph with the darkest blue color represents the maximum observed GC content values in the specific region, the graph in the middle with the intermediate blue color shows the mean observed GC content values in the specific region, and the graph at the bottom with the light blue color shows the minimum observed GC content values in the specific region.

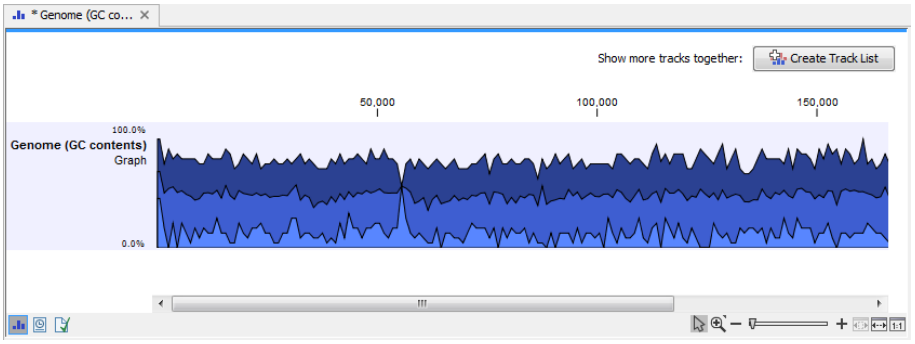


Figure 19.19: *The output from "Create GC Content Graph" is a graph track. The graph track shows one value for each base with one graph being available for each chromosome.*

When zooming all the way in to single nucleotide level only one graph can now be seen as you ar now no longer looking at large genomic regions. Instead, you can now use the tooltip by mousing over each individual base to look at the GC content for that particular base and the number of bases that you specified as the window size to be used. This is shown in figure 19.20 where the top part of the figure shows the graph track when zoomed all the way out and the bottom part of the figure shows a genome browser view with the sequence track that was used as input together with the output graph track. The input and the output tracks were combined in one view as a track list (see section 19.3) by clicking on the button labeled **Create Genome Browser View** found in the upper right corner of the graph track in the top part of the figure (see the red arrow).

This track can then be displayed together with the sequence and other tracks in a genome browser view.

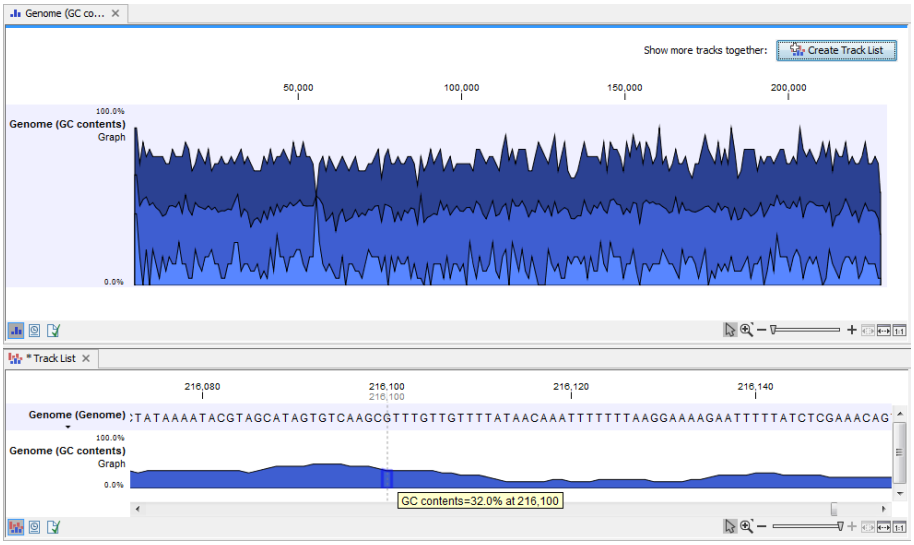


Figure 19.20: *The top part of the figure shows the graph track when zoomed all the way out. The bottom part of the window shows a graph track together with the input genomic sequence at single nucleotide resolution. By mousing over one nucleotide, you can see the GC content for this position. In our example we chose a window size of 25 nucleotides and the GC content that is shown for one nucleotide is the GC content for the central nucleotide and the 12 bases upstream and downstream of this nucleotide.*

19.4.2 Create Mapping Graph

The **Create Mapping Graph** tool can create a range of different graphs from a read mapping track. To run the tool go to the toolbox:

Toolbox | **Genome Browser** ( ) | **Graphs** | **Create Mapping Graph**

Select the read mapping as shown in figure 19.21 and click on the button labeled **Next**.

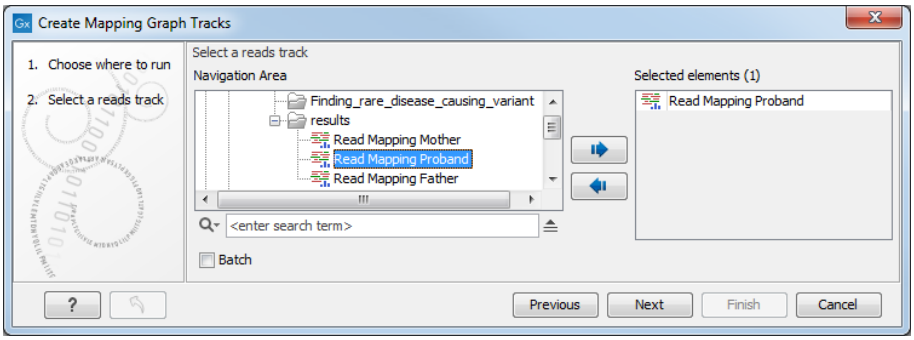


Figure 19.21: *Creating graph track from mappings.*

Select the graph tracks that you would like to create.

One graph track output will be created for each of the graph tracks you have chosen by checking the boxes shown in figure 19.22.

The following options exist:

- Read coverage. For each position this graph shows the number of reads contributing to the alignment (see a more elaborate definition in section 20.2.2).

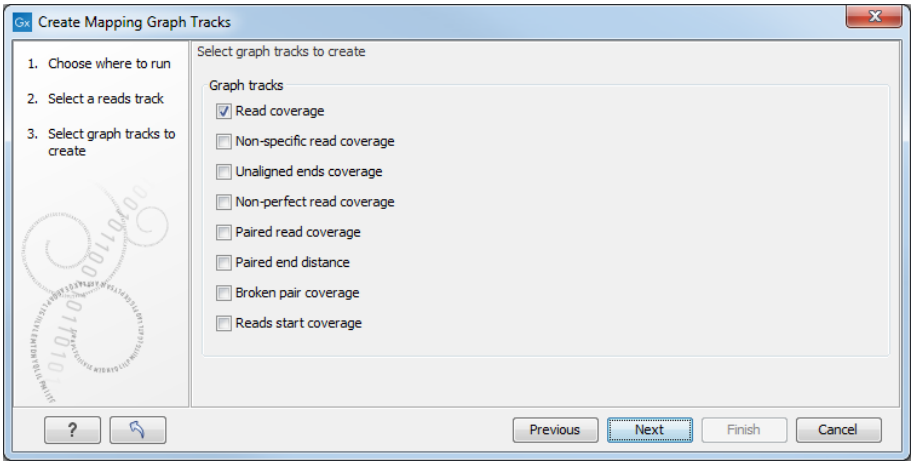


Figure 19.22: *Choose the types of graph tracks you would like to generate.*

- Non-specific read coverage. Non-specific reads are reads that would fit equally well other places in the reference genome.
- Unaligned ends coverage. Un-aligned ends arise when a read has been locally aligned to a reference sequence, and then end of the read is left unaligned because there are mismatches or gaps relative to the reference sequence. This part of the read does not contribute to the read coverage above. The unaligned ends coverage graph shows how many reads that have unaligned ends at each position.
- Non-perfect read coverage. Non-perfect reads are reads with one or more mismatches or gaps relative to the reference sequence.
- Paired read coverage. This lists the coverage of intact pairs. Coverage is counted as one in the overlapping region.
- Paired end distance. Displays the average distance between the forward and the reverse read in a pair.
- Broken pair coverage. A pair is broken either because only one read in the pair matches, or because the distance or relative orientation between the reads is wrong.
- Reads start coverage. For each position this graph shows the number of reads that start in that position.

Click on the button labeled **Next**, choose where to save the generated output(s) and click on the button labeled **Finish**.

An example of three different outputs is shown in figure 19.23. Two of the views have been dragged and dropped to other areas of the **View Area** to be able to see them in the same window. If you would like to learn more about how to do this, please refer to section 2.1.6.

In this example we generated all possible outputs and chose to open them without saving. You can see that the names of the tabs are marked with an asterisk, which indicates that the graph shown in the view area has not been saved or that changes have been made that must be saved if you want to keep them. Three of the generated outputs have been opened. If you would like to see the outputs in the same view, you can do this by creating a genome browser view. Click on the button labeled **Create genome Browser View** in the upper right corner of each of the graph

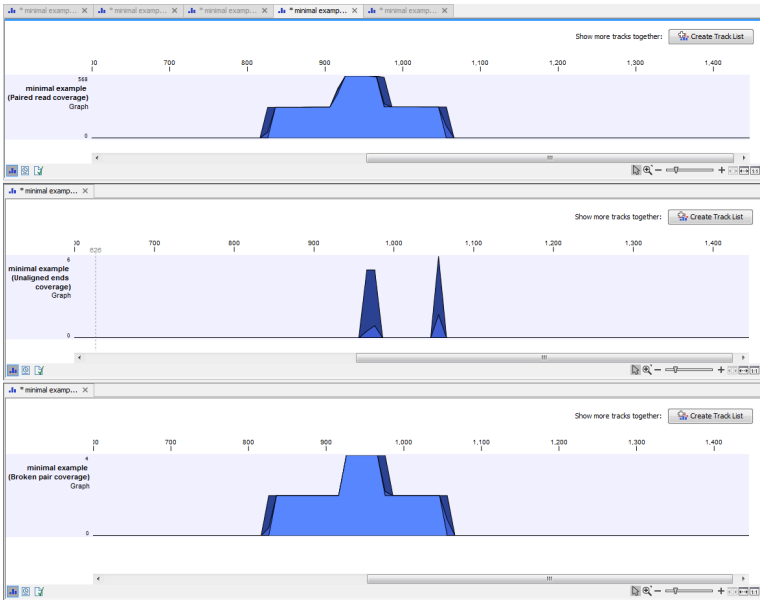


Figure 19.23: *Three types of graph tracks are shown.*

tracks shown in the **View Area**. Combining graph tracks in a genome browser view links the individual tracks together, which makes it much easier to compare the different graph tracks.

Note that the option "Fix graph bounds" found under **Track layout** in the **Side Panel** is useful to manually adjust the numbers on the y-axis.

19.4.3 Identify Graph Threshold Areas

The **Identify Graph Threshold Areas** tool uses graph tracks as input to identify graph regions that fall within certain limits or thresholds. Both a lower and an upper threshold can be specified to create an annotation track for those regions of a graph track where the values are in the given range (see figure 19.24). The range chosen for the lower and upper thresholds will depend on the data (coverage, quality etc).

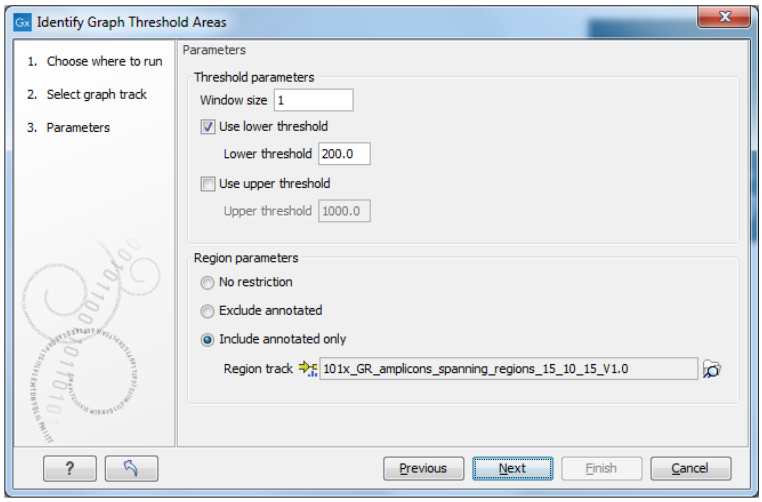


Figure 19.24: *Specification of lower and upper thresholds.*

The **window-size** parameter specifies the width of the window around every position that is used

to calculate an average value for that position and hence "smoothes" the graph track beforehand. A window size of 1 will simply use the value present at every individual position and determine if it is within the upper and lower threshold, hence resulting in the same "non-smoothing" behavior as previous versions of the workbench without this parameter. In contrast, a window size of 100 checks if the average value derived from the surrounding 100 positions falls between the minimum and maximum threshold. Such larger windows help to prevent "jumps" in the graph track from fragmenting the output intervals or help to detect over-represented regions in the track that are only visible when looked at in the context of larger intervals and lower resolution.

It is also possible to restrict the tool to certain regions.

An example output is shown in figure 19.25 where the coverage graph has a couple of local minima near zero. However, by using the averaging window, the tool is able to produce a single unbroken annotation covering the entire region. Of course larger window sizes result in regions that are broader and hence their boundaries are less likely to exactly coincide with the borders of visually recognizable borders of regions in the track.

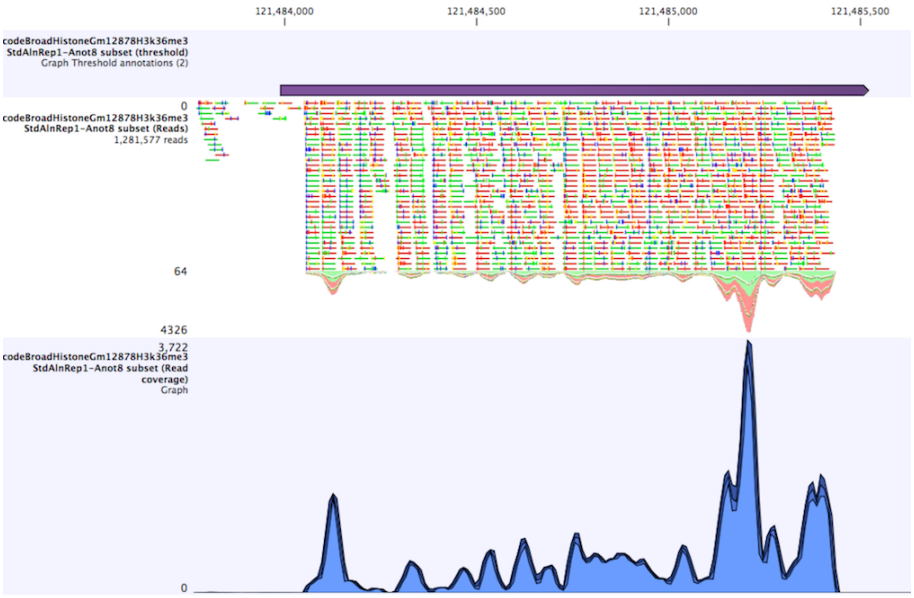


Figure 19.25: *Track list including a region identified by the parameters set above on a dataset of H3K36 methylation from ENCODE. The top track shows the resulting region. Below is the track containing the reads. The graph track at the bottom shows the coverage with the minimum, mean, and maximum observed values.*

When zoomed out, the graph tracks are composed of three curves showing the maximum, mean, and minimum value observed in a given region (see figure 19.25). When zoomed in all the way down to base resolution only one curve will be shown reflecting the exact observation at each individual position.

Part V

Initial data handling

Chapter 20

Quality control tools

Contents

20.1 QC for Target Sequencing

This tool is designed to report the performance (enrichment and specificity) of a targeted re-sequencing experiment. Targeted re-sequencing is due to its low costs, very popular and several companies provide platforms and protocols (learn more at http://en.wikipedia.org/wiki/Exome_sequencing#Target-enrichment_strategies). Array-based approaches are offered by e.g. Agilent (SureSelect) and Roche Nimblegen. Furthermore, amplicon sequencing with PCR primers is offered by RainDance, Fluidigm and others.

Given an annotation track with the target regions (e.g. imported from a bed file), this tool will investigate a read mapping to determine whether the targeted regions have been appropriately covered by sequencing reads as well as information about how specific the reads map to the targeted regions. The results are provided both as a summary report and as track or table with detailed information about each targeted region.

20.1.1 Running the "QC for Target Sequencing" tool

The tool is found in the Toolbox:

Toolbox | **Quality Control** ( ) | **QC for Target Sequencing** ( )

This opens a wizard where you can select mapping results ( )/ ( )/ ( ). Clicking **Next** will take you to the wizard shown in figure 20.1.

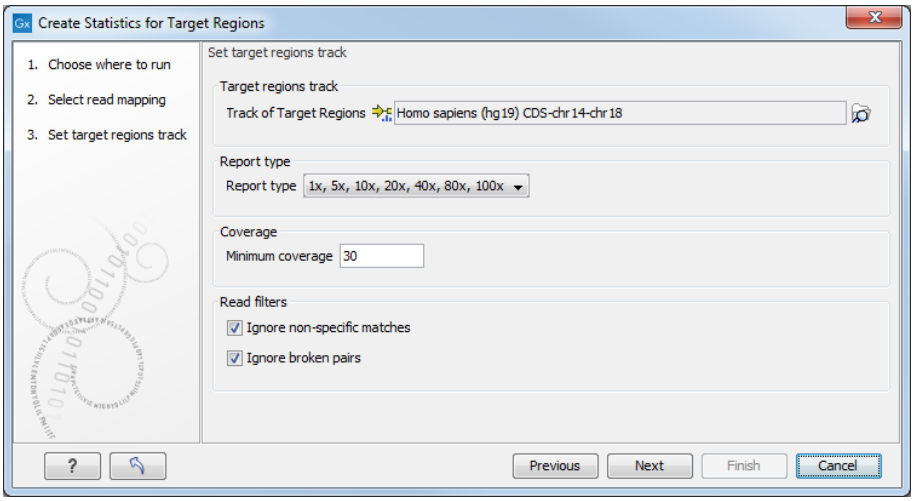


Figure 20.1: *Specifying the track of target regions.*

Click the **Browse** ( ) icon to select an annotation track that defines the targeted regions of your reference genome.

The **Report type** allows you to select different sets of predefined coverage thresholds to use for reporting (see below). Furthermore, you will be asked to provide a **Minimum coverage** threshold, i.e., the minimum coverage needed on all positions in a target, in order for that target to be considered covered.

Finally, you are asked to specify whether you want to **Ignore non-specific matches** and **Ignore broken pairs**. When these are applied reads that are non-specifically mapped or belong to broken pairs will be ignored.

Click **Next** to specify the type of output you want (see figure 20.2).

There are three options:

- The report gives an overview of the whole data set as explained in section 20.1.2.
- The track gives information on coverage for each target region as described in section 20.1.3.
- The coverage table outputs coverage for each position in all the targets as described in section 20.1.4.
- The coverage graph outputs a graphical presentation of the coverage for each position in all the targets. Positions outside the targets will have the value 0. The values are calculated by the "Target regions statistics" tool - that is, where broken pairs and multi-hit reads are included or ignored, depending upon what the user has specified in the wizard. On the x-axis is the reference position; on the y-axis is the coverage. The x-axis and y-axis values are identical to those found in the corresponding columns of the coverage table.

Click **Finish** to create the selected reports.

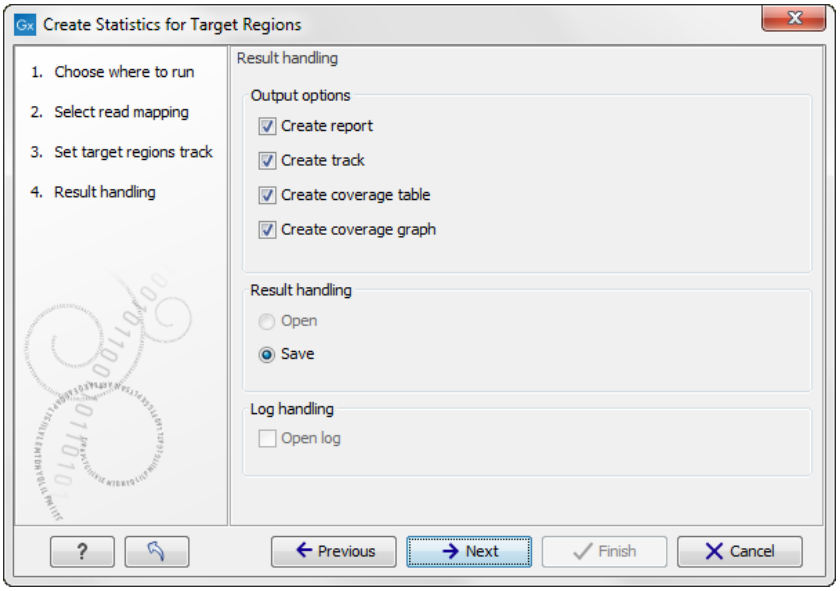


Figure 20.2: *Specifying how the result should be reported.*

20.1.2 Coverage summary report

An example of a coverage report is shown in figure 20.3).

This figure shows only the top of the report. The full content is explained below:

Coverage summary This table shows overall coverage information.

Number target regions The number of targeted regions.

Total length of target regions The sum of the size of all the targeted regions (this means it is calculated from the annotations alone and is not influenced by the reads).

Average coverage For each position in each target region the coverage is calculated, and stored (you can see the individual coverages in the **Coverage table** output, figure 20.8). The 'average coverage' is calculated by taking the mean of all the calculated coverages in all the positions in all target regions. Note that if the user has chosen **Ignore non-specific matches** or **Ignore broken pairs** these reads will not contribute to the coverage. Note also that bases in over-lapping paired reads will only be counted as 1.

Number of target regions with low coverage The number of target regions which have positions with a coverage that is below the user-specified **Minimum coverage** threshold.

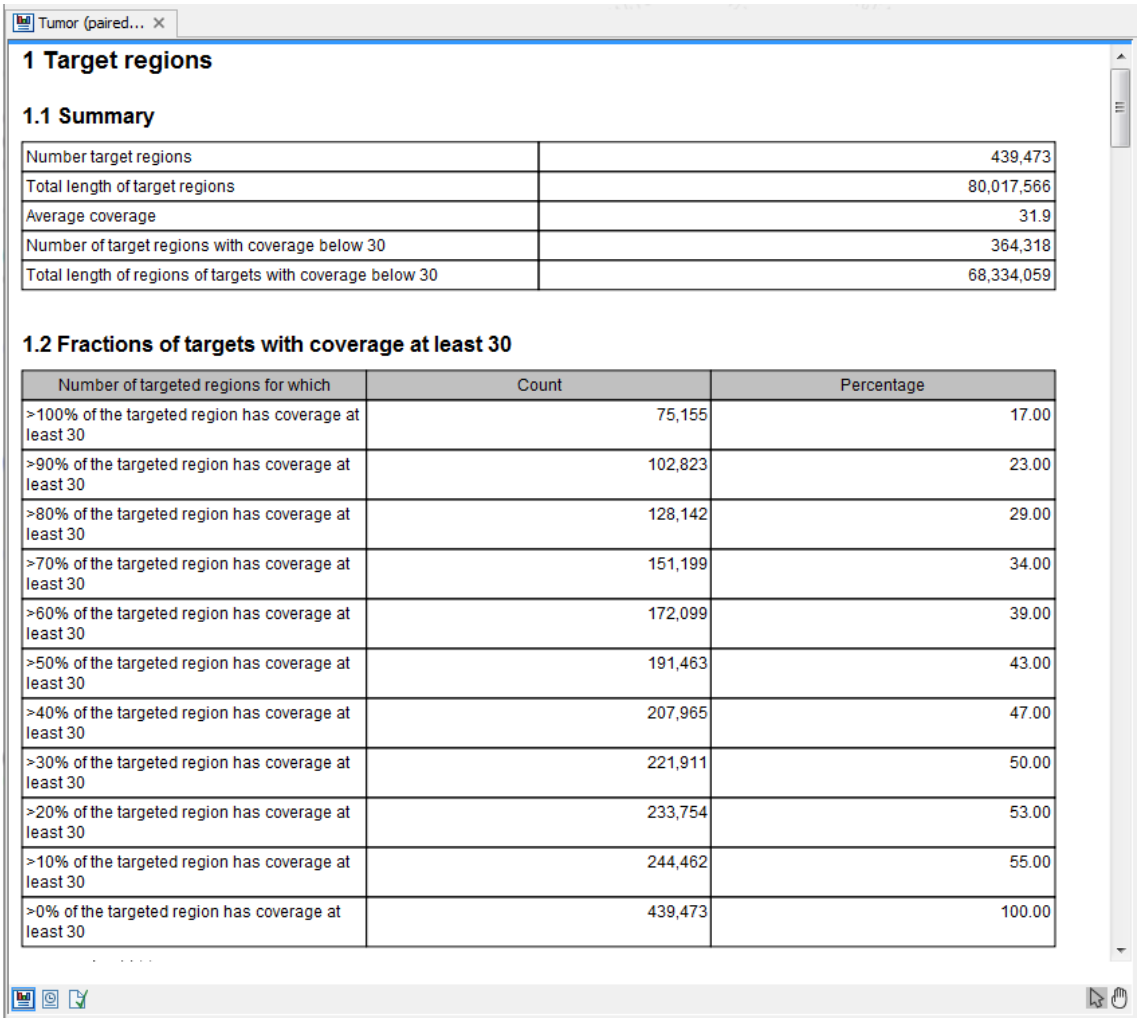
Total length of target regions with low coverage The total length of these regions.

Fractions of targets with coverage at least... This table shows how many target regions have a certain percentage of the region above the user-specified **Minimum coverage** threshold.

Fractions of targets with coverage at least... A histogram presentation of the table above in **Fractions of targets with coverage at least...**.

Coverage of target regions positions This plot shows the coverage level on the x axis, and the number of positions in the target regions with that coverage level.

Coverage of target regions positions A version of the histogram above zoomed in to the values that lie +- 3SDs from the median.


Tumor (paired... ×

1 Target regions

1.1 Summary

Number target regions	439,473
Total length of target regions	80,017,566
Average coverage	31.9
Number of target regions with coverage below 30	364,318
Total length of regions of targets with coverage below 30	68,334,059

1.2 Fractions of targets with coverage at least 30

Number of targeted regions for which	Count	Percentage
>100% of the targeted region has coverage at least 30	75,155	17.00
>90% of the targeted region has coverage at least 30	102,823	23.00
>80% of the targeted region has coverage at least 30	128,142	29.00
>70% of the targeted region has coverage at least 30	151,199	34.00
>60% of the targeted region has coverage at least 30	172,099	39.00
>50% of the targeted region has coverage at least 30	191,463	43.00
>40% of the targeted region has coverage at least 30	207,965	47.00
>30% of the targeted region has coverage at least 30	221,911	50.00
>20% of the targeted region has coverage at least 30	233,754	53.00
>10% of the targeted region has coverage at least 30	244,462	55.00
>0% of the targeted region has coverage at least 30	439,473	100.00

Figure 20.3: *The report with overviews of mapped reads.*

Minimum coverage of target regions This shows the percentage of the targeted regions that are covered by this many bases. The intervals can be specified in the dialog when running the analysis. Default is 1, 5, 10, 20, 40, 80, 100 times. In figure 20.4 this means that 26.58 % of the positions on the target are covered by at least 40 bases.

Targeted regions overview This section contains two tables: one that summarizes, for each reference sequence, information relating to the *reads* mapped, and one that summarizes, for each reference, information relating to the *bases* mapped (figures 20.4 and 20.5). Note that, for the table that is concerned with *reads*, reads in over-lapping pairs are counted individually. Also note that, for the table that is concerned with *bases*, bases in overlapping paired reads are counted only as one (Examples are given in figures 20.6 and figure 20.7).

Reference The name of the reference sequence.

Total mapped reads The total number of mapped reads on the reference, including reads mapped outside the target regions.

Mapped reads in targeted region Total number of reads in the targeted regions. Note that if there are overlapping regions, reads covered by two regions will be counted twice. If a read is only partially inside a targeted region, it will still count as a full read.

Specificity The percentage of the total mapped reads that are in the targeted regions.

Total mapped reads excl ingored The total number of mapped reads on the reference, including reads mapped outside the target regions, excluding the non-specific matches or broken pairs, if the user has switched on the option to ignore those.

Mapped reads in targeted region excl ingored Total number of reads in the targeted regions, excluding the non-specific matches or broken pairs, if the user has switched on the option to ignore those.

Specificity excl ingored The percentage of the total mapped reads that are in the targeted regions.

Reference The name of the reference sequence.

Total mapped bases The total number of mapped bases on the reference, including bases mapped outside the target regions.

Mapped bases in targeted region Total number of bases mapped within in the targeted regions. Note that if there are overlapping regions, bases included in two regions will be counted twice.

Specificity The percentage of the total mapped bases that are in the targeted regions.

Total mapped bases excl ingored The total number of mapped bases on the reference, including bases mapped outside the target regions, excluding the bases in non-specific matches or broken pairs, if the user has switched on the option to ignore those.

Mapped bases in targeted region excl ingored Total number of bases in the targeted regions, excluding the bases in non-specific matches or broken pairs, if the user has switched on the option to ignore those.

Specificity excl ingored The percentage of the total mapped bases that are in the targeted regions.

Distribution of target region length A plot of the length of the target regions, and a version of the plot where only the target region lengths that lie within +3SDs of the median target length are shown.

Base coverage The percentage of base positions in the target regions that are covered by respectively 0.1, 0.2, 0.3, 0.4, 0.5 and 1.0 times the mean coverage, where the mean coverage is the **average coverage** given in table 1.1. Because this is based on mean coverage, the numbers can be used for cross-sample comparison of the quality of the experiment.

Base coverage plot A plot showing the relationship between fold mean coverage and the number of positions. This is a graphical representation of the **Base coverage** table above.

Mean coverage per target position Three plots listing the mean coverage for each position of the targeted regions. The first plot shows coverage across the whole target, using a percentage of the target length on the x axis (to make it possible to have targets with different lengths in the same plot). This is reported for reverse and forward reads as well. In addition, there are two plots showing the same but with base positions on the x axis counting from the start and end of the target regions, respectively. These plots can be used to evaluate whether there is a general tendency towards lower coverage at the end of the targeted region, and whether there is a bias in terms of forward and reverse reads coverage.

Read count per %GC The plot shows the GC content of the reference sequence on the X-axis and the number of mapped reads on the Y-axis. This plot will show if there is a basis caused by higher GC-content in the sequence.

1.5 Minimum coverage of target regions

Coverage	
1 x	95.06%
5 x	89.83%
10 x	82.40%
20 x	62.40%
40 x	26.58%
80 x	4.78%
100 x	2.34%

2 Targeted region overview

Reference	Total mapped reads	Mapped reads in targeted region	Specificity (%)	Total mapped reads excl ignored	Mapped reads in targeted region excl ignored	Specificity excl ignored (%)
1	4,338,296	2,168,439	49.98	4,338,296	2,168,439	49.98
2	4,929,239	2,384,422	48.37	4,929,239	2,384,422	48.37
3	2,495,513	1,242,395	49.79	2,495,513	1,242,395	49.79
4	2,214,025	768,447	34.71	2,214,025	768,447	34.71

Figure 20.4: *The report: mapped reads.*

Reference	Total mapped bases	Mapped bases in targeted region	Specificity (%)	Total mapped bases excl ignored	Mapped bases in targeted region excl ignored	Specificity excl ignored (%)
1	344,955,568	252,700,911	73.26	344,955,568	252,700,911	73.26
2	388,499,642	276,219,534	71.10	388,499,642	276,219,534	71.10
3	196,369,744	144,978,039	73.83	196,369,744	144,978,039	73.83
4	173,524,865	91,208,669	52.56	173,524,865	91,208,669	52.56

Figure 20.5: *The report: mapped bases.*

20.1.3 Per-region statistics

In addition to the summary report, you can see coverage statistics for each targeted region. This is reported as a track, and you can see the numbers by going to the table ( ) view of the track. An example is shown in figure 20.6:

Chromosome The name is taken from the reference sequence used for mapping.

Region The targeted region.

Name The annotation name derived from the annotation (if there is additional information on the annotation, this is retained in this table as well).

Target region length The length of the region.

Target region length with coverage above... The length of the region that is covered by at least the **Minimum coverage** level provided in figure 20.1.

Percentage with coverage above... The percentage of the positions in the region with coverage at least the **Minimum coverage** level provided in figure 20.1.

Read count Number of reads that cover this region. Note that reads that only cover the region partially are also included. Note that reads in over-lapping pairs are counted individually (see figures 20.6 and figure 20.7).

Base count The number of bases in the reads that are covering the target region. Note that bases in overlapping pairs are counted only once (see figures 20.6 and figure 20.7).

%GC The GC content of the region.

Min coverage The lowest coverage in the region.

Max coverage The highest coverage in the region.

Mean coverage The average coverage in the region.

Median coverage The median coverage in the region.

Zero coverage bases The number of positions with no coverage.

Mean coverage (excluding zero coverage) The average coverage in the region, excluding any zero-coverage parts.

Median coverage (excluding zero coverage) The median coverage in the region, excluding any zero-coverage parts.

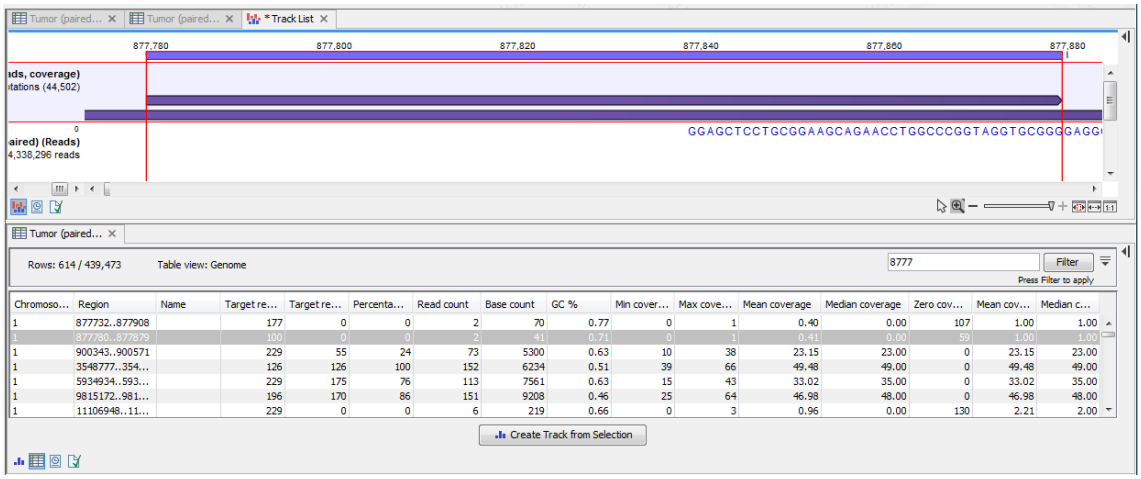


Figure 20.6: *A track list containing the target region coverage track and reads track. The target region coverage track has been opened from the track list and is shown in table view. Detailed information on each region is displayed. Only one paired read maps to the region selected.*

In the shown figure, the coverage table is shown in split view with a track list. The coverage track has been opened from the track list by clicking once on the name found in the left side in the track list. When opening a read mapping in split view from a track list, the two views are linked,

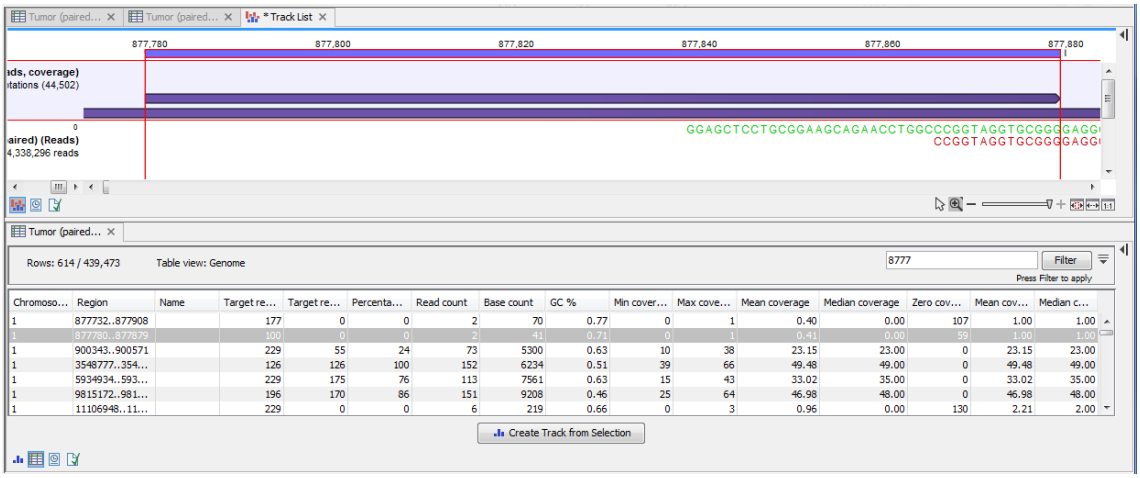


Figure 20.7: *The same data as shown in figure 20.6, but now the Disconnect paired reads option in the side-panel of the reads track has been ticked, so that the two reads in the paired read are shown disconnected.*

this means that when you click on an entry in the table, this position will be brought into focus in the track list.

For information about how to create a track list, please see section 19.3.

20.1.4 Coverage table

Besides standard information such as position etc, the coverage table (figure 20.8) lists the following information for each position in the whole target:

Name The name of the target region.

Target region position The name of the target region.

Reference base The base in the reference sequence.

Coverage The number of bases mapped to this position. Note that bases in over-lapping pairs are counted only once. Also note that if the user has chosen the **Ignore non-specific matches** or **Ignore broken pairs** options, these reads will be ignored. (see discussion on coverage in section 20.2.2).

In the shown figure, the coverage table is shown in split view with a track list. The coverage track has been opened from the track list by clicking once on the name found in the left side in the track list. When opening a read mapping in split view from a track list, the two views are linked, this means that when you click on an entry in the table, this position will be brought into focus in the track list.

For information about how to create a track list, please see section 19.3.

20.1.5 Coverage graph

The coverage graph is a graphical presentation of the coverage for each position in all the targets (positions outside the targets will have the value 0). The values are calculated by the "Target

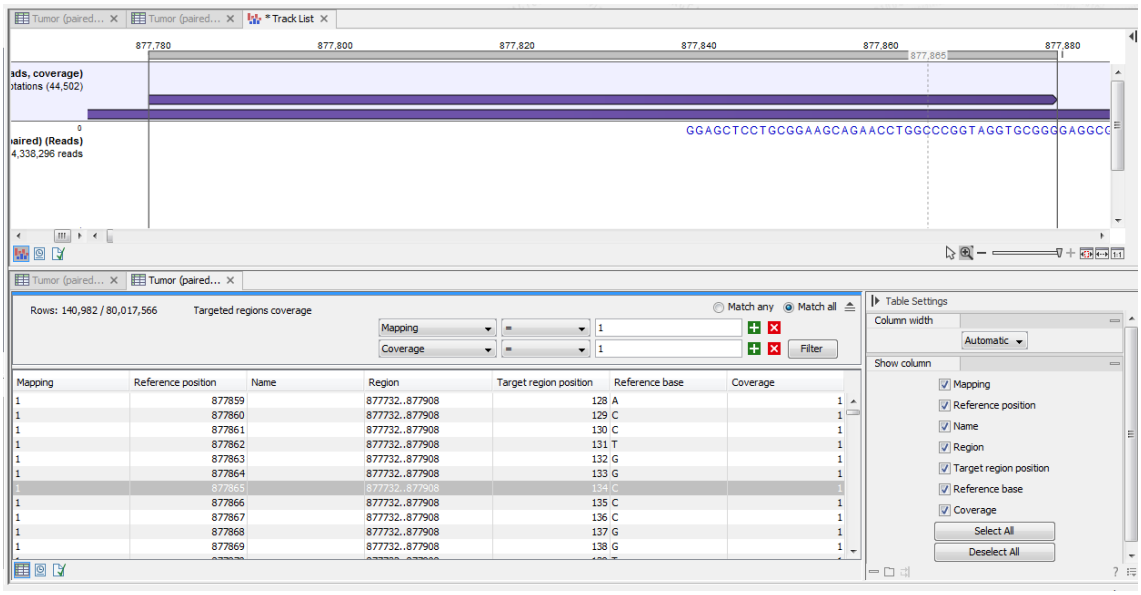


Figure 20.8: *The targeted region coverage table for the same region as shown in same as shown in figures 20.6 and figure 20.7.*

regions statistics" tool, and are presented with the reference position on the x-axis and the coverage on the y-axis (see figure 20.9). The x-axis and y-axis values are identical to those found in the columns of the coverage table.

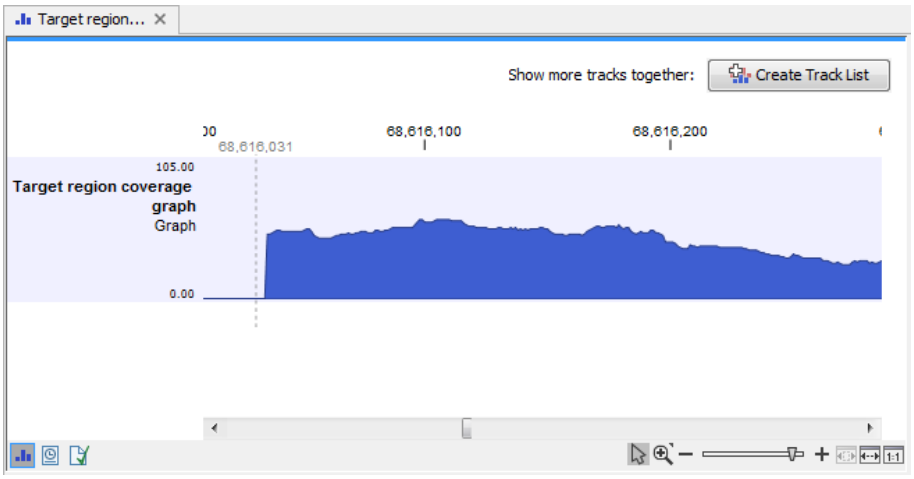


Figure 20.9: *An example of a targeted region coverage graph.*

20.2 QC for Sequencing Reads

Quality assurance as well as concern regarding sample authenticity in biotechnology and bioengineering have always been serious topics in both production and research. While next generation sequencing techniques greatly enhance in-depth analyses of DNA-samples, they, however, introduce additional error-sources. Resulting error-signatures can neither be easily removed from resulting sequencing data nor even recognized, which is mainly due to the massive amount of data. Altogether biologists and sequencing facility technicians face not only issues of minor relevance, e.g. suboptimal library preparation, but also serious incidents, including sample-contamination or even mix-up, ultimately threatening the accuracy of biological conclusions.

Unfortunately, most of the problems and evolving questions raised above can't be solved and answered entirely. However, the sequencing data quality control tool of the *Biomedical Genomics Workbench* provides various generic tools to assist in the quality control process of the samples by assessing and visualizing statistics on:

- Sequence-read lengths and base-coverages
- Nucleotide-contributions and base-ambiguities
- Quality scores as emitted by the base-caller
- Over-represented sequences and hints suggesting contamination events

This tool aims at assessing above quality-indicators and investigates proper and improper result presentation. The inspiration comes from the FastQC-project (`http://www.bioinformatics.babraham.ac.uk/projects/fastqc/`).

20.2.1 QC Sequencing Report Content

The report comes in two different flavors: a supplementary report consisting of tables representing all the values that are calculated, and a main summary graphical report where the tables are visualized in plots (see an example in figure 20.10). Both reports can be exported as pdf files or Excel spread sheets.

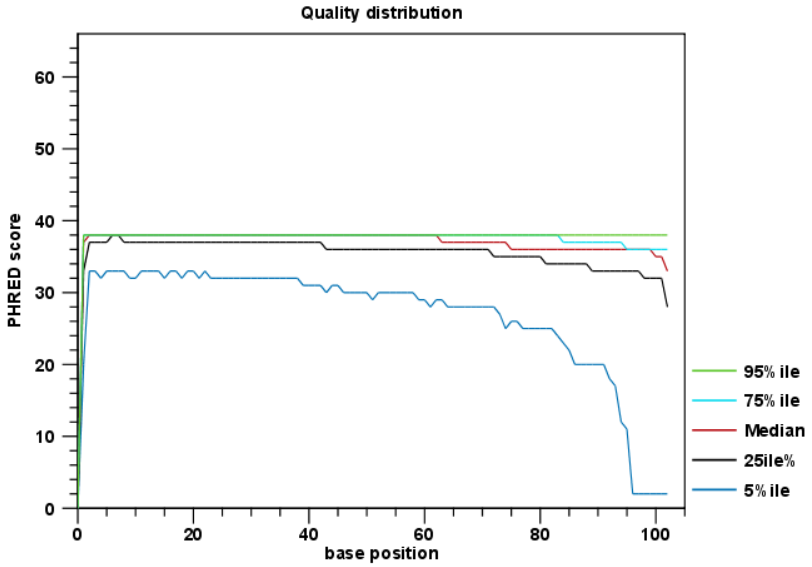


Figure 20.10: *An example of a plot from the graphical report, showing the quality values per base position.*

The Summary table provides information regarding the creation date, the author, the software used, the number of data sets the report is based upon, as well as data name and content in terms of read number and total number of nucleotides. The report is then divided in per-sequence and per-base analyses. In the per-sequence analyses, some characteristic (a single value) is assessed for each sequence and then contributes to the overall assessment. In per-base assessments each base position is examined and counted independently. In both sections, the first items

assess the most simple characteristics that are supported by all sequencing technologies while the quality analyses examine quality scores reported from technology-dependent base callers. Please note that the NGS import tools of the *CLC Genomics Workbench* and *CLC Genomics Server* convert quality scores to PHRED-scale, regardless of the data source.

Per-sequence analysis

Lengths distribution Counts the number of sequences that have been observed for individual sequence lengths. The resulting table correlates sequence-lengths in base-pairs with numbers of sequences observed with that number of base-pairs. The length distribution depends on your library preparation and sequencing protocol. If you observe secondary peaks at unexpected lengths you may want to consider removing these. Using the Workbench Trim tool you can trim away reads above and/or below a certain length.

GC-content distribution Counts the number of sequences that feature individual %GC-contents in 101 bins ranging from 0 to 100%.The %GC-content of a sequence is calculated by dividing the absolute number of G/C-nucleotides by the length of that sequence, and should look like a normal distribution in the range of what is expected for the genome you are working with. If the GC-content is substantially lower (the normal distribution is shifted to the left), it may be that GC-rich areas have not been properly covered. You can check this by mapping the reads to your reference. A non-normal distribution, or one that has several peaks indicates the presence of contaminants in the reads.

Ambiguous base content Counts the number of sequences that feature individual %N-contents in 101 bins ranging from 0 to 100%, where N refers to all ambiguous base-codes as specified by IUPAC.The %N-content of a sequence is calculated by dividing the absolute number of ambiguous nucleotides through the length of that sequence. This distribution should be as close to 0 as possible.

Quality distribution Calculates the amount of sequences that feature individual PHRED-scores in 64 bins from 0 to 63. The quality score of a sequence as calculated as arithmetic mean of its base qualities. PHRED-scores of 30 and above are considered high quality. If you have many reads with low quality you may want to discuss this with your sequencing provider. Low quality bases/reads can also be trimmed off with the Trim Reads tool.

Per-base analysis

Coverage Calculates absolute coverages for individual base positions. The resulting graph correlates base-positions with the number of sequences that supported (covered) that position.

Nucleotide contributions Calculates absolute coverages for the four DNA nucleotides (A, C, G or T) for each base position in the sequences. In a random library you would expect little or no difference between the bases, thus the lines in this plot should be parallel to each other. The relative amounts of each base should reflect the overall amount of the bases in your genome. A strong bias along the read length where the lines fluctuate a lot for certain positions may indicate that an over-represented sequence is contaminating your sequences. However, if this is at the 5' or 3' ends, it will likely be adapters that you can remove using the Trim Reads tool.

GC-content Calculates absolute coverages of C's + G's for each base position in the sequences. If you see a GC bias with changes at specific base positions along the read length this could indicate that an over-represented sequence is contaminating your library.

Ambiguous base-content Calculates absolute coverages of N's, for each base position in the sequences, where N refers to all ambiguous base-codes as specified by IUPAC.

Quality distribution Calculates the amount of bases that feature individual PHRED-scores in 64 bins from 0 to 63. This results in a three-dimensional table, where dimension 1 refers to the base-position, dimension 2 refers to the quality-score and dimension 3 to amounts of bases observed at that position with that quality score. PHRED-scores above 20 are considered good quality. It is normal to see the quality dropping off near the end of reads. Such low-quality ends can be trimmed off using the Trim Reads tool.

Over-representation analyses

Enriched 5-mer distribution The 5-mer analysis examines the enrichment of penta-nucleotides. The enrichment of 5-mers is calculated as the ratio of observed and expected 5-mer frequencies. The expected frequency is calculated as product of the empirical nucleotide probabilities that make up the 5-mer. (Example: given the 5-mer = CCCCC and cytosines have been observed to 20% in the examined sequences, the 5-mer expectation is 0.2^5). Note that 5-mers that contain ambiguous bases (anything different from A/T/C/G) are ignored. This analysis calculates the absolute coverage and enrichment for each 5-mer (observed/expected based on background distribution of nucleotides) for each base position, and plots position vs enrichment data for the top five enriched 5-mers (or fewer if less than five enriched 5-mers are present). It will reveal if there is a bias at certain positions along the read length. This may originate from non-trimmed adapter sequences, poly A tails and more.

Sequence duplication levels The duplicated sequences analysis identifies sequence reads that have been sequenced multiple times. A high level of duplication may indicate an enrichment bias, as for instance introduced by PCR amplification. Please note that multiple input sequence lists will be considered as one federated data set for this analysis. Batch mode can be used to generate separate reports for individual sequence lists.

In order to identify duplicate reads the tool examines all reads in the input and uses a clone dictionary containing per clone the read representing the clone and a counter representing the size of the clone. For each input read these steps are followed: (1) check whether the read is already in the dictionary. (2a) if yes, increment the according counter and continue with next read. (2b) if not, put the read in the dictionary and set its counter to 1.

To achieve reasonable performance, the dictionary has a maximum capacity of 250,000 clones. To this end, step 2a involves a random decision as to whether a read is granted entry into the clone dictionary. Every read that is not already in the dictionary has the same chance T of entering the clone dictionary with T = 250,000 / total amount of input reads. This design has the following properties:

- The clone dictionary will ultimately contain at most 250,000 entries.
- The sum of all clone sizes in the dictionary amounts at most to the total number of input reads.

- Because of T being constant for all input reads, even a cluster of reads belonging to the same clone and first occurring towards the end of the input can be detected.
- Because of the random sampling, the tool might underestimate the size of a read clone, specifically if its first read representative does not make it into the dictionary. The ratio is that a larger clone has a higher cumulative chance of being eventually represented in the dictionary than a smaller clone.

Because all current sequencing techniques tend to report decreasing quality scores for the 3' ends of sequences, there is a risk that duplicates are NOT detected, merely because of sequencing errors towards their 3' ends. The identity of two sequence reads is therefore determined based on the identity of the first 50nt from the 5' end.

The results of this analysis are presented in a plot and a corresponding table correlating the clone size (duplication count) with the number of clones of that size. For example, if the input contains 10 sequences and each sequence was seen exactly once, then the table will contain only one row with duplication-count=1 and sequence-count=10. Note: due to space restrictions the corresponding bar-plot shows only bars for duplication-counts of x=[0-100]. Bar-heights of duplication-counts >100 are accumulated at x=100. Please refer to the table-report for a full list of individual duplication-counts.

Duplicated sequences This results in a list of actual sequences most prevalently observed. The list contains a maximum of 25 (most frequently observed) sequences and is only present in the supplementary report.

20.2.2 Adapters

Currently, adapter contamination, i.e. adapter sequences in the reads, cannot be detected in a reliable way with this tool. In some cases, adapter contamination will show up as enriched 5-mers near the end of sequences but only if the contamination is severe.

20.2.3 Running the "QC for Sequencing Reads" tool

The tool is found in the Toolbox:

Toolbox | **Quality Control** () | **QC for Sequencing Reads** ()

Select one or more sequence lists with sequencing reads as input. If sequence lists in the Navigation Area were already selected, these will be shown in the Selected Elements window. When multiple lists are selected as an input, they are all analyzed in one pool. If you need separate reports for each data set, you can run it in a batch. Click **Next** to adjust output options which allow you to select the graphical and supplementary report.

20.3 QC for Read Mapping

You can create two kinds of reports regarding read mappings: *First*, you can choose to generate a summary report about the mapping process itself (see section 22.2.2). *Second*, you can generate a detailed statistics report after the mapping has finished. This report is useful if you want to generate statistics across results made in different processes, and it generates more detailed statistics than the summary mapping report. Both reports are described below.

20.3.1 Running the "QC for Read Mapping" tool

The tool is found in the Toolbox:

Toolbox | **Quality Control** ( ) | **QC for Read Mapping** ( )

This opens a dialog where you can select mapping results ( )/ ( )/ ( ) or RNA-Seq analysis results ( ).

Clicking **Next** will display the dialog shown in figure 20.11

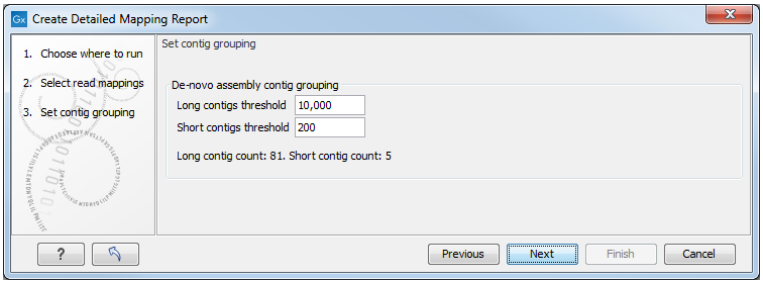


Figure 20.11: *Parameters for mapping reports.*

The next wizard step shows the used thresholds for the mapping report. These parameters cannot be modified by the user (as thresholds can only be specified for de novo assemblies that do not have a consensus sequence).

Click **Next** to select output options as shown in figure 20.12

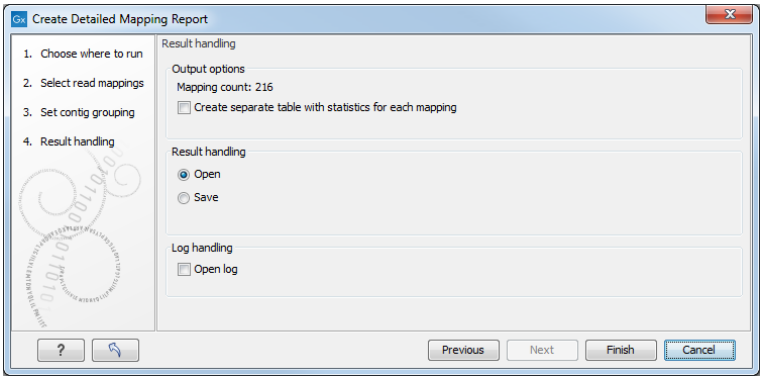


Figure 20.12: *Result handling options.*

By checking "Create table with statistics for each mapping", you can create a table showing detailed statistics for each reference sequence.

The first section of the detailed report is a summary of the statistics: reference count, type, total reference length, GC contents in %, total read count, mean read length, and total read length

- Reference count
- Type
- Total reference length
- GC contents in %

- Total read count
- Mean read length
- Total read length

The rest of the report, as well as the optional statistic tables are described in the following sections.

References

The second section of the detailed report concerns the Reference sequence(s).

First, a table gives information about **Reference coverage**, including coverage statistics and GC content of the reference sequence.

The second table gives **Coverage statistics**. A position on the reference is counted as "covered" when at least one read is aligned to it. Note that unaligned ends (faded nucleotides at the ends) that are produced when mapping using local alignment do not contribute to the coverage. Also, positions with an ambiguous nucleotide in the reference (i.e., not A, C, T or G) count as zero coverage regions, regardless of the number of reads mapping across them.

In the example shown in figure 20.13, there is a region of zero coverage in the middle and one time coverage on each side. Note that the gaps to the very right are within the same read which means that these two positions on the reference sequence are still counted as "covered".

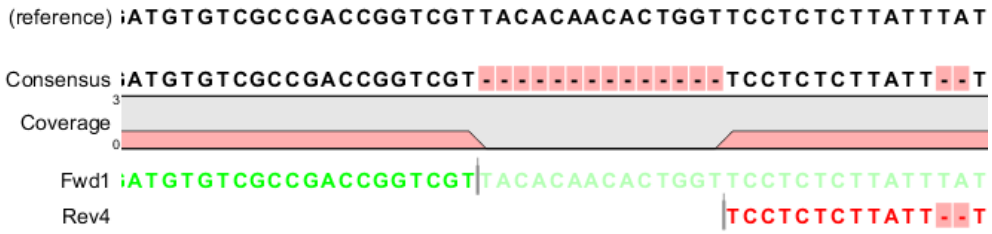


Figure 20.13: *A region of zero coverage in the middle and one time coverage on each side. Note that the gaps to the very right are within the same read which means that these two positions on the reference sequence are still counted as "covered".*

In this table, coverage is reported on two levels: including and excluding zero coverage regions. In some cases, you do not expect the whole reference to be covered, and only the coverage levels of the covered parts of the reference sequence are interesting. On the other hand, if you have sequenced the full genome that you use as reference, the overall coverage is probably the most relevant number (i.e. including zero coverage regions).

In the third and fourth subsections, two graphs display **Coverage level distribution**, with and without zero coverage regions. Two bar plots show the distribution of coverage with coverage level on the x-axis and number of positions with that coverage on the y-axis (as seen in figure 20.14).

The graph to the left shows all the coverage levels, whereas the graph to the right shows coverage levels within 3 standard deviations from the mean. The reason for this is that for

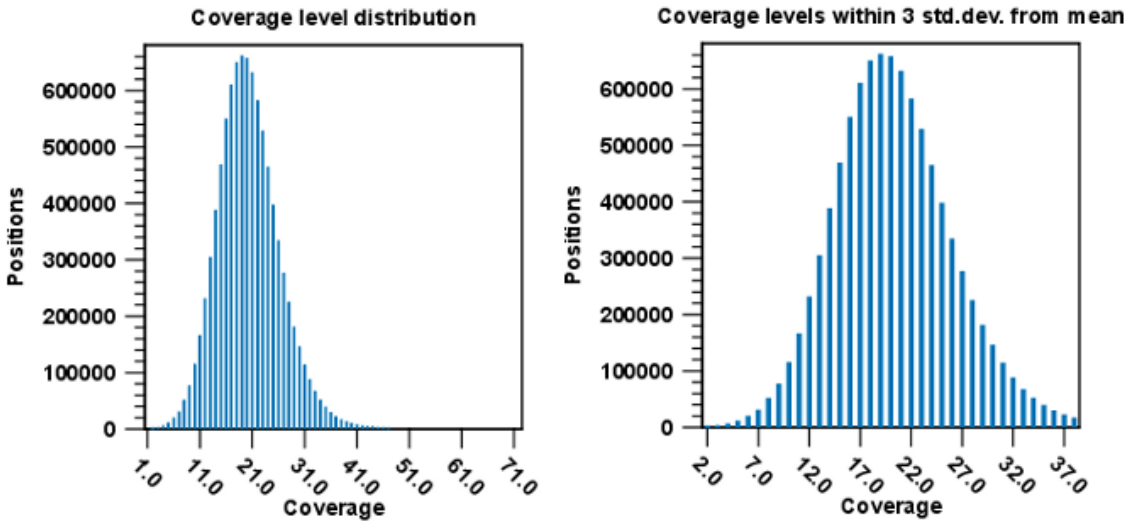


Figure 20.14: *Distribution of coverage - to the left for all the coverage levels, and to the right for coverage levels within 3 standard deviations from the mean.*

complex genomes, you will often have a few regions with extremely high coverage which will affect the resolution of the graph, making it impossible to see the coverage distribution for the majority of the references. These coverage outliers are excluded when only showing coverage within 3 standard deviations from the mean. Below the second coverage graph there are some statistics on the data that is outside the 3 standard deviations.

Subsection 5 gives some statistics on the **Zero coverage regions**; the number, minimum and maximum length, mean length, standard deviation, and total length.

One of the biases seen in sequencing data concerns GC content. Often there is a correlation between GC content and coverage. In order to investigate this correlation, the report includes in subsection 6 a **Coverage versus GC Content** graph plotting coverage against GC content (see figure 20.15). Note that you can see the GC content for each reference sequence in the table(s) above.

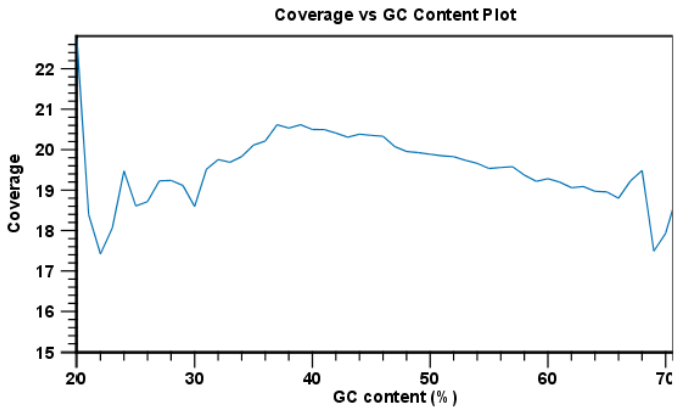


Figure 20.15: *The plot displays, for each GC content level (0-100 %), the mean read coverage of 100bp reference segments with that GC content.*

The plot displays, for each GC content level (0-100 %), the mean read coverage of 100bp reference segments with that GC content.

For a report created from a de novo assembly, this section finishes with statistics about the reads which are the same for both reference and de novo assembly (see section 20.2.2 below).

Read statistics

This section contains simple statistics for **all mapped reads**, **non-specific matches** (reads that match more than place during the assembly), **non-perfect matches** (reads with one or more mismatches or gaps relative to the reference sequence) and **paired reads**.

Note! Paired reads are counted as two, even though they form one pair. The section on paired reads also includes information about paired distance and counts the number of pairs that were broken due to:

- Wrong distance: When starting the mapping, a distance interval is specified. If the reads during the mapping are placed outside this interval, they will be counted here.
- Mate inverted: If one of the reads has been matched as reverse complement, the pair will be broken (note that the pairwise orientation of the reads is determined during import).
- Mate on other contig: If the reads are placed on different contigs, the pair will also be broken.
- Mate not matched: If only one of the reads match, the pair will be broken as well.

Each subsection contains a table that recapitulates the read count, % of all mapped reads, mean read length and total read length, and for some sections two graphs showing the distribution of match specificity or the distribution of mismatches.

Note that for the section concerning paired reads (see figure 20.16), the distance includes both the read sequence and the insert between them as explained in section 6.3.7.

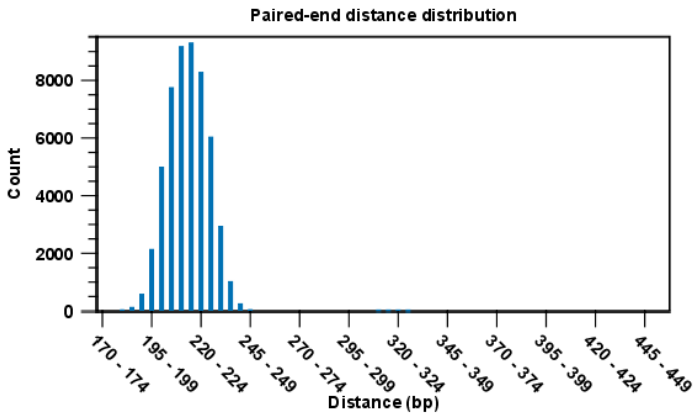


Figure 20.16: *A bar plot showing the distribution of distances between intact pairs.*

The following subsections give graphs showing **read length distribution**, **insertion length distribution**, **deletion length distribution**. Two plots of the distribution of insertion and deletion lengths can be seen in figure 20.17 and figure 20.18.

Nucleotide differences in reads relative to a reference gives the percentage of read bases that differ with the reference for all base pairs and a deletion. In the **Nucleotide mapping** section two tables give the counts and percentages of differences between the reads and the reference for each base. Graphs display the relative errors and errors counts between reads to reference and reference to reads, i.e., which bases in the reference are substituted to which bases in the reads. This information is plotted in different ways with an example shown here in figure 20.17.

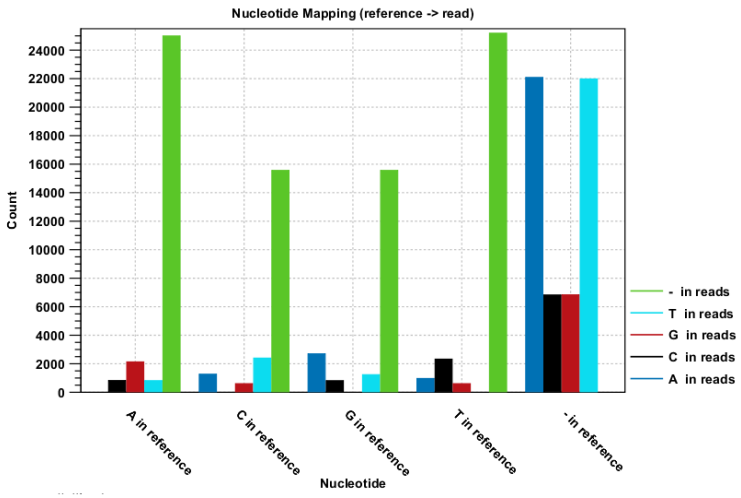


Figure 20.17: *The As and Ts are more often substituted with a gap in the sequencing reads than C and G.*

This example shows for each type of base in the reference sequence, which base (or gap) is found most often. Please note that only mismatches are plotted - the matches are not included. For example, an A in the reference is more often replaced by a G than any other base.

Below these plots, there are two plots of the **quality values for matches** and **quality values for mismatches**. Next, there is a plot of the mismatch fraction for each read position. Typically with quality dropping towards the end of a read, there will be more mismatches towards the end as the example in figure 20.18 shows.

The last plots section deals with unaligned read lengths.

Statistics table for each mapping

By checking "Create table with statistics for each mapping", a table showing detailed statistics for each reference sequence will be generated (figure 20.19).

- **Contig**
- **Reference name**
- **Reference Latin name**
- **Reference description**
- **Reference length**

Figure 20.18: *There are mismatches towards the end of the reads.*

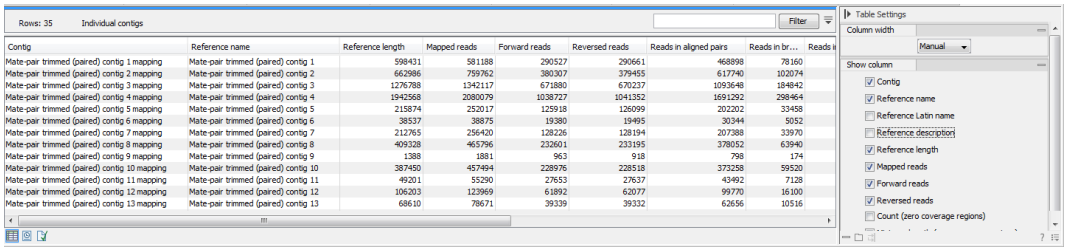


Figure 20.19: *Statistics table for a read mapping.*

- **Mapped reads**
- **Forward reads**
- **Reversed reads**
- **Reads in aligned pairs**
- **Reads in broken pairs: wrong distance or mate inverted**
- **Reads in broken pairs: mate on other contig**
- **Reads in broken pairs: mate not mapped**
- **Average distance**
- **Standard deviation distance**. Standard deviation of the mapped pairs distances.
- **Non-specific matches**
- **Non-perfect matches**
- **Minimum coverage**
- **Maximum coverage**
- **Average coverage**

- **Standard deviation coverage**. Standard deviation of the per base coverage.
- **Minimum coverage excluding zero coverage regions**
- **Average excluding zero coverage regions**
- **Standard deviation excluding zero coverage regions**. Standard deviation of the per base coverage, excluding regions without coverage.
- **% GC**. GC content of the reference sequence.
- **Consensus length**
- **Fraction of reference covered**
- **Count (zero coverage regions)**
- **Minimum length (zero coverage regions)**
- **Maximum length (zero coverage regions)**
- **Average length (zero coverage regions)**
- **Standard deviation length (zero coverage regions)**. Standard deviation of the distribution of the lengths of all the zero coverage regions on that contig.
- **Total length (zero coverage regions)**

Chapter 21

Preparing raw data tools

Contents

21.1 Merge overlapping pairs

Some paired end library preparation methods using relatively short fragment size will generate data with overlapping pairs. This type of data can be handled as standard paired-end data in the *Biomedical Genomics Workbench*, and it will work perfectly fine (see details for variant detection in section 22.21).

However, in some situations it can be useful to merge the overlapping pair into one sequence read instead. The benefit is that you get longer reads, and that the quality improves (normally the quality drops towards the end of a read, and by overlapping the ends of two reads, the consensus read now reflects two read ends instead of just one).

In the *Biomedical Genomics Workbench*, there is a tool for merging overlapping reads, which are in forward-reverse orientation:

Toolbox | **NGS Core Tools** () | **Merge Overlapping Pairs** ()

Select one or more sequence lists with paired end sequencing reads as input.

Please note that read pairs have to be in forward-reverse orientation. Please also note that after merging the merged reads will always be in the forward orientation. As an aside, length

trimming of reads can be done before or after merging, however the merged read's 3' is the 5' of the reverse read, so that trimming the original reads using the **Remove 5' terminal nucleotides** option corresponds to trimming the merged reads using both the **Remove 5' terminal nucleotides** option and the **Remove 3' terminal nucleotide**s option.

Clicking **Next** allows you to set parameters as displayed in figure 21.1.

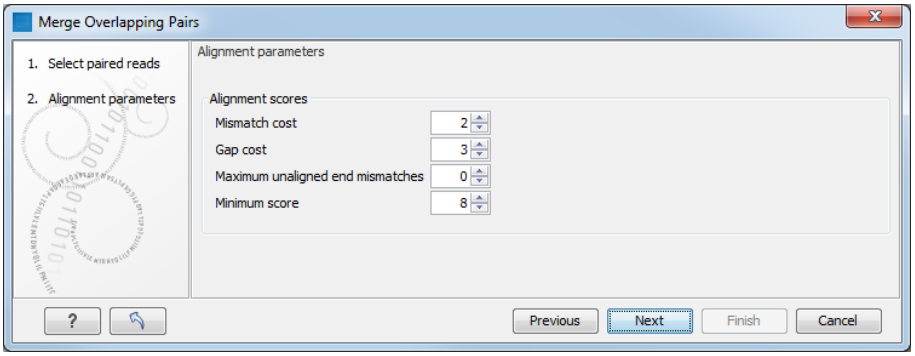


Figure 21.1: *Setting parameters for merging overlapping pairs.*

In order to understand how these parameters should be set, an explanation of the merging algorithm is needed: Because the fragment size is not an exact number of base pairs and is different from fragment to fragment, an alignment of the two reads has to be performed. If the alignment is *good and long enough*, the reads will be merged. *Good enough* in this context means that the alignment has to satisfy some user-specified score criteria (details below). Because of sequencing errors that typically are more abundant towards the end of the read, the alignment is not expected always to be perfect, and the user can decide how many errors are acceptable. *Long enough* in this context means that the overlap between the reads has to be non-coincidental. Merging two reads that do not really overlap, leads to errors in the downstream analysis, thus it is very important to make sure that the overlap is big enough. If only a few bases overlap was required, some read pairs will match by chance, so this has to be avoided.

The following parameters are used to define what is *good enough* and *long enough*

- **Mismatch cost** The alignment awards one point for a match, and the mismatch cost is set by this parameter. The default value is 2.
- **Gap cost** This is the cost for introducing an insertion or deletion in the alignment. The default value is 3.
- **Max unaligned end mismatches** The alignment is local, which means that a number of bases can be left unaligned. If the quality of the reads is dropping to be very poor towards the end of the read, and the expected overlap is long enough, it makes sense to allow some unaligned bases at the end. However, this should be used with great care which is why the default value is 0. As explained above, a wrong decision to merge the reads leads to errors in the downstream analysis, so it is better to be conservative and accept fewer merged reads in the result.
- **Minimum score** This is the minimum score of an alignment to be accepted for merging. The default value is 10. As an example: with default settings, this means that an overlap of 13 bases with one mismatch will be accepted (12 matches minus 2 for a mismatch).

Please note that even with the alignment scores above the minimum score specified in the tool setup, the paired reads also need to have the number of end mismatches below the "Maximum unaligned end mismatches" value specified in the tool setup to be qualified for merging.

After clicking **Next** you can select whether a report should be generated as part of the output. The main result will be two sequence lists for each list in the input: one containing the merged reads (marked as single end reads), and one containing the reads that could not be merged (still marked as paired data). Since the *Biomedical Genomics Workbench* handles a mix of paired and unpaired data, both of these sequence lists can be used in the further analysis. However, please note that low quality can be one of the reasons why a pair cannot be merged. Hence, the list of reads that could not be paired is more likely to contain more reads with errors than the one with the merged reads.

21.1.1 Using quality scores when merging

Quality scores come into play in two different ways when merging overlapping pairs.

First, if there is a conflict between the reads in a pair (i.e. a mismatch or gap in the alignment), quality scores are used to determine which base the merged read should have at a given position. The base with the highest quality score will be the one used. In the case of gaps, the average of the quality scores of the two surrounding bases will be used. In the case that two conflicting bases have the same quality or both reads have no quality scores, an [IUPAC ambiguity code](see the appendix section G) representing these bases will be inserted.

Second, the quality scores of the merged read reflect the quality scores of the input reads.

We assume independence of errors in calculating the new quality score for a merged position and carry out the following calculations:

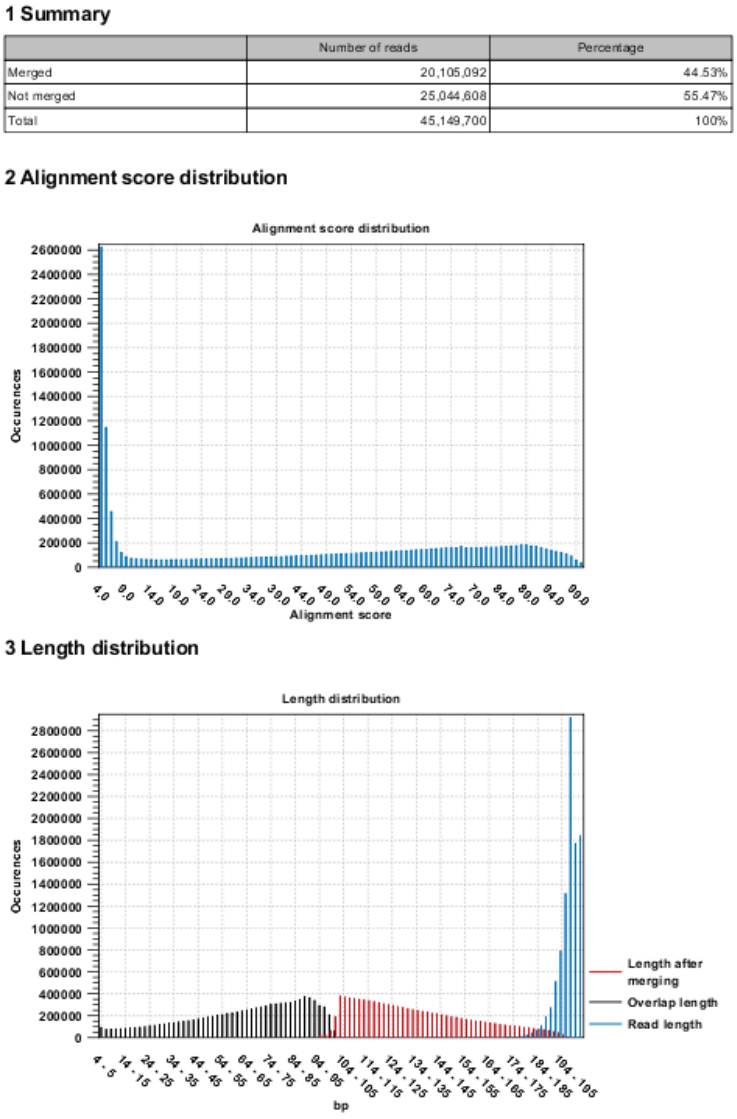
- When the two reads agree at a position, the two quality scores are summed to form the quality score of the base in the new read. The score is capped at the maximum value on the quality score scale which is 64. Phred scores are log scores, so their sums represent the multiplication of the original error probabilities.
- If the two bases disagree at a position, the quality score of the base in the new read is determined by subtracting the lowest score from the highest score of the input reads. Similar to the addition of scores when bases are the same, this adjusts the error probability to reflect a decreased certainty that the base reported at that position is correct.

Thus, if two bases at a given position of an overlapping region are different, and each of those bases was originally given a high phred score, the score assigned to the merged base will be very low. This reflects the fact that the base at this position is unreliable.

If a base at a given position in one read of an overlapping region has a very low quality score and the base at that position in the other read has a high score, it is likely that the base with the high quality score is correct. The adjusted quality score for this position in the merged read would reflect that there is less certainty in the base at that position than before. However, such a position would still be assigned quite a high quality, as the base call is still likely to be correct.

21.1.2 Report of merged pairs

Figure 21.2 shows an example of the report generated when merging overlapping pairs.



1 Summary

	Number of reads	Percentage
Merged	20,105,092	44.53%
Not merged	25,044,608	55.47%
Total	45,149,700	100%

2 Alignment score distribution

3 Length distribution

Figure 21.2: *Report of overlapping pairs.*

It contains three sections:

- A summary showing the numbers and percentages of reads that have been merged.
- A plot of the alignment scores. This can be used to guide the choice of minimum alignment score as explained in section 21.1.
- A plot of read lengths. This shows the distribution of read lengths for the pairs that have been merged:
  - The length of the overlap.
  - The length of the merged read.
  - The combined length of the two reads in the pair before merging.

21.2 Trim Reads

Biomedical Genomics Workbench offers a number of ways to trim your sequence reads prior to assembly and mapping, including adapter trimming, quality trimming and length trimming. For each original read, the regions of the sequence to be removed for each type of trimming operation are determined independently according to choices made in the trim dialogs. The types of trim operations that can be performed are:

1. Quality trimming based on quality scores
2. Ambiguity trimming to trim off stretches of Ns for example
3. Adapter trimming (automatic, or also with a Trim Adapter List, see section 21.2.2)
4. Base trim to remove a specified number of bases at either 3' or 5' end of the reads
5. Length trimming to remove reads shorter or longer than a specified threshold

The trim operation that removes the largest region of the original read from either end is performed while other trim operations are ignored as they would just remove part of the same region.

Note that this may occasionally expose an internal region in a read that has now become subject to trimming. In such cases, trimming may have to be done more than once.

The result of the trim is a list of sequences that have passed the trim (referred to as the trimmed list below) and optionally a list of the sequences that have been discarded and a summary report (list of discarded sequences). The original data will not be changed.

To start trimming:

Toolbox | **Tools** | **Preparing Raw Data** () | **Trim Reads** ()

This opens a dialog where you can add sequences or sequence lists. If you add several sequence lists, each list will be processed separately and you will get a a list of trimmed sequences for each input sequence list.

When the sequences are selected, click **Next**.

21.2.1 Quality trimming

This opens the dialog displayed in figure 21.3 where you can specify parameters for quality trimming.

The following parameters can be adjusted in the dialog:

- **Trim using quality scores.** If the sequence files contain quality scores from a base-caller algorithm this information can be used for trimming sequence ends. The program uses the modified-Mott trimming algorithm for this purpose (Richard Mott, personal communication):

  Quality scores in the Workbench are on a Phred scale, and formats using other scales will be converted during import. The Phred quality scores (Q), defined as: $Q = -10log10(P)$, where P is the base-calling error probability, can then be used to calculate the error probabilities, which in turn can be used to set the limit for, which bases should be trimmed.

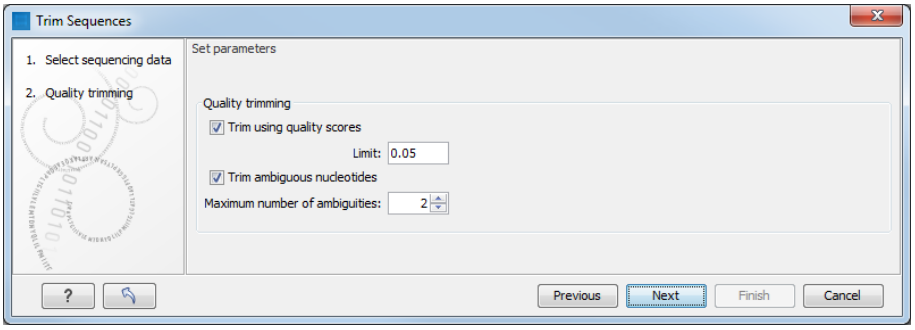


Figure 21.3: *Specifying quality trimming.*

Hence, the first step in the trim process is to convert the quality score (Q) to an error probability: $p_{error} = 10^{\frac{Q}{-10}}$. (This now means that low values are high quality bases.)

Next, for every base a new value is calculated: $Limit - p_{error}$. This value will be negative for low quality bases, where the error probability is high.

For every base, the Workbench calculates the running sum of this value. If the sum drops below zero, it is set to zero. The part of the sequence not trimmed will be the region ending at the highest value of the running sum and starting at the last zero value before this highest score. Everything before and after this region will be trimmed. A read will be completely removed if the score never makes it above zero.

- **Trim ambiguous nucleotides.** This option trims the sequence ends based on the presence of ambiguous nucleotides (typically N). Note that the automated sequencer generating the data must be set to output ambiguous nucleotides in order for this option to apply. The algorithm takes as input the *maximal number of ambiguous nucleotides allowed in the sequence after trimming*. If this maximum is set to e.g. 3, the algorithm finds the maximum length region containing 3 or fewer ambiguities and then trims away the ends not included in this region. The "Trim ambiguous nucleotides" option trims all types of ambiguous nucleotides (see Appendix G).

21.2.2 Adapter trimming

Clicking **Next** will allow you to specify parameters for adapter trimming.

When you are analyzing sequencing data, the adapters must be trimmed off before you proceed with further analysis. The removal of adapters is often done directly on the sequencing machine, but in some cases, some adapters remain on the sequenced reads. The presence of remaining adapters can lead to misleading results, so we recommend to trim them off the reads (figure 21.4).

The default option for this trimming step is to use the "Automatic read-through adapter trimming", which will detect read-through adapter sequence on paired-end reads automatically. Read-through means that the sample DNA fragment being sequenced is shorter than the read length, such that the 3' end of one read includes the reverse-complement of the adapter from the start of the other read. Leaving this option enabled is always recommended: the trimming performed automatically can detect read-through of even a single nucleotide, which is not the case when trimming using a trim adapter list. The detected adapters for the first and second read can be found in the Trim Reads report.

There are however a couple of limitations on the "Automatic read-through adapter trimming" option: this option detects overlap in paired reads containing standard nucleotides (A, T, C,

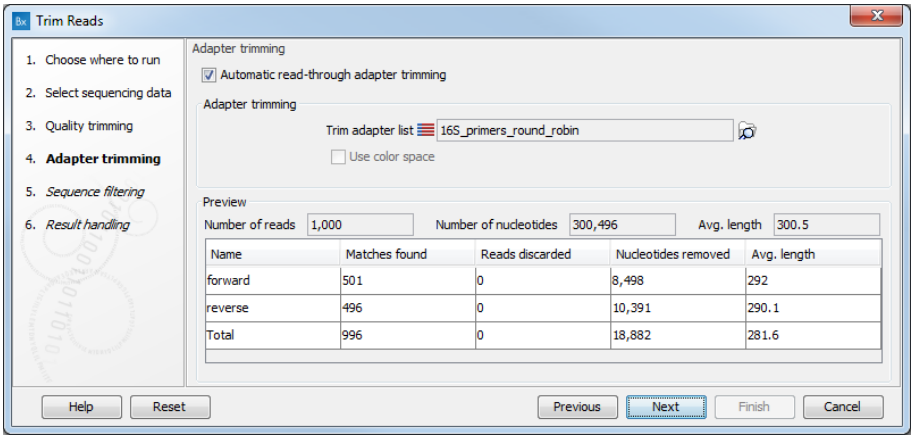


Figure 21.4: *Trimming your sequencing data for adapter sequences.*

and G). If the read contains ambiguous symbols, such as N, these will not match the standard nucleotides.

Also, the first and second read should be of equal (or near-equal) length - some sequencing protocols use asymmetric read lengths for the first and second read, in which case the tool is less likely to detect and trim the read-through.

So when you are working with data of low quality, asymmetric read lengths, mate-paired reads, single reads, small RNAs, or also when working with gene specific primers, it is recommended that you specify a trim adapter read in addition to using the "Automatic read-through adapter trimming" option. It is even possible to use the report of the Trim Read tool to find out what Trim adapter list should be used for the data at hand. Read section 21.2.3 to learn how to create an adapter list.

You can specify if the adapter trimming should be performed in **Color space**. Note that this option is only available for sequencing data imported using the SOLiD import (see section 36.2). When doing the trimming in color space, the Smith-Waterman alignment is simply done using colors rather than bases. The adapter sequence is still input in base space, and the Workbench then infers the color codes. The scoring thresholds apply to the color space alignment (this means that a perfect match of 10 bases would get a score of 9 because 10 bases are represented by 9 color residues). Learn more about color space in section 22.4.

Below you find a preview listing the results of trimming with the adapter trimming list on 1000 reads in the input file (reads 1001-2000 when the read file is long enough). This is useful for a quick feedback on how changes in the parameters affect the trimming (rather than having to run the full analysis several times to identify a good parameter set). The following information is shown:

- **Name**. The name of the adapter.
- **Matches found**. Number of matches found based on the settings.
- **Reads discarded**. This is the number of reads that will be completely discarded. This can either be because they are completely trimmed (when the **Action** is set to Remove adapter and the match is found at the 3' end of the read), or when the **Action** is set to Discard when found or Discard when not found.
- **Nucleotides removed**. The number of nucleotides that are trimmed include both the ones

coming from the reads that are discarded and the ones coming from the parts of the reads that are trimmed off.

- **Avg. length** This is the average length of the reads that are retained (excluding the ones that are discarded).

Note that the preview panel is only showing how the trim adapter list will affect the results. Other kinds of trimming (automatic trimming of read-through adapters, quality or length trimming) are not reflected in the preview table.

21.2.3 Trim adapter list

The Trim Reads tool is set by default to detect automatically read-through adapters present in the reads used as input for the tool. We recommend to always enable this option. In addition, you can use a Trim adapter list for a more through trimming of read-through adapter in reads of lower quality, or to trim for specific adapters that are not read-through such as small RNAs, gene specific primers, or when working with single or mate-paired reads. In such cases, you have to import or create a Trim adapter list that must be supplied to the Trim Reads tool.

Creating a new Trim adapter list

It is possible to generate a Trim adapter list directly in the workbench. Go to:

File | **New** | **Trim Adapter List**

This will create a new empty Trim adapter list. At the bottom of the view, you have the following options that allow you to edit the Trim adapter list:

- **Add Rows.** Add a new adapter.
- **Edit Row**. Edit the selected adapter. This can also be achieved by double-clicking the relevant row in the table.
- **Delete Row**. Delete the selected adapter.

Add the adapter(s) that you would like to use for trimming by clicking on the button **Add Row** ( ) found at the bottom of the View Area. Adding an adapter is done in two steps. In the first wizard step (figure 21.5), you enter the basic information about the adapter, and how the trimming should be done relative to the adapter found.

In the second dialog (figure 21.6), you define the scores that will be used to recognize adapters. For each read sequence in the input, a Smith-Waterman alignment [Smith and Waterman, 1981] is carried out with each adapter sequence. Alignment scores are computed and compared to the minimum scores provided for each adapter when setting up the Trim adapter List. If the alignment score is higher or equal to the minimum score, the adapter is recognized and the trimming can happen as specified in the first wizard. If however the alignment score is lower than the minimum score, the adapter is not recognized and trimmed.

Trim adapter

Start by providing the name and sequence of the adapter that should be trimmed away. Use the **Reverse Complement** button to reverse complement the sequence you typed in if it is found

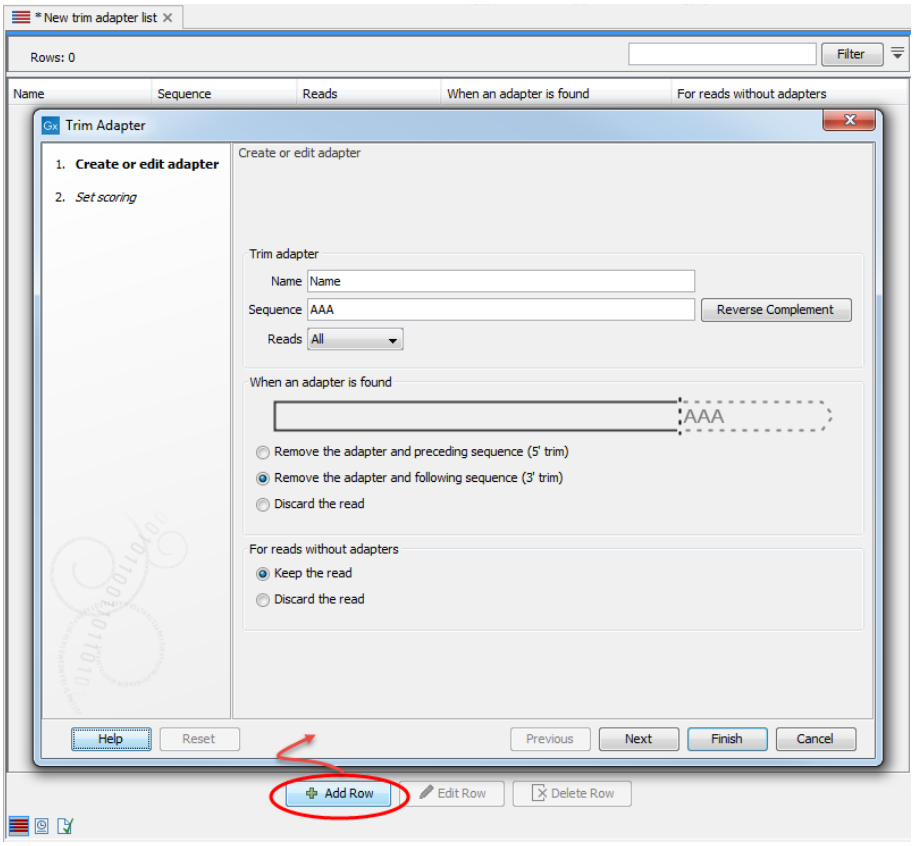


Figure 21.5: *Add an adapter to the Trim Adapter List by clicking on the button labeled "Add Row" found at the bottom of the New Trim Adapter view.*

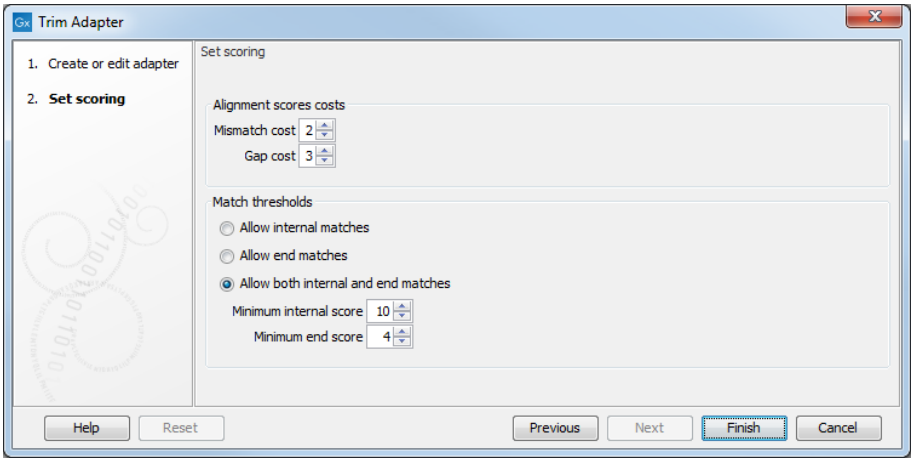


Figure 21.6: *Set the scoring used to define what will be considered as adaptor.*

in reverse complement in the reads. You can then specify whether you want the adapter to be trimmed on all reads, or more specifically on the first or second read of a pair.

When an adapter is found

Once you have entered the sequence of the adapter, a visual shows how the adapter will be trimmed, allowing you to decide which option suits your needs best:

- Remove the adapter and preceding sequence (5’ trim)

- Remove the adapter and following sequence (3' trim)
- Discard the read. The read will be placed in the list of discarded sequences. This can be used for quality checking the data for linker contamination for example.

For reads without adapters

You can decide here what to do with reads where no adapter was found. This kind of adapter trimming is particularly useful for small RNA sequencing where the remnants of the adapter is an indication that this is indeed a small RNA. Beware of lists where multiple adapters have been set to "Discard the read" when the adapters are not found: only sequences containing **all** the adapters will remain in the list of trimmed reads.

Alignment scores costs

An A,C,G or T in the adapter that matches an A,C,G or T respectively - or a corresponding ambiguity code letter - in a sequence is considered a match and will be awarded 1 point. However, you can decide how much penalty should be awarded to mismatches and gaps:

- **Mismatches** The penalty for mismatches between bases is set as 2 by default.
- **Gap** The penalty for gaps introduced into the alignment is set as 3 by default.

Here are the few examples of adapter matches and corresponding scores (figure 21.7). These examples are all internal matches where the alignment of the adapter falls within the read.

```
    CGTATCAATCGATTACGCTATGAATG
a)     ||||||| ||||            11 matches - 2 mismatches = 7
       TTCAATCGGTTAC

    CGTATCAATCGATTCGCTATGAATG
       |||||||||| ||||         14 matches - 1 gap = 11
b)     ATCAATCGAT-CGCT

    CGTATCAATCGATTACGCTATGAATG
c)     |||||||                 7 matches - 3 mismatches = 1
       TTCAATCGGG
```

Figure 21.7: *Three examples showing a sequencing read (top) and an adapter (bottom). The examples are artificial, using default setting with mismatch costs = 2 and gap cost = 3.*

Match thresholds

Note that there is a difference between an **internal match** and an **end match**. An end match happens when the alignment of the adapter starts at the end of the sequence that is being trimmed. This can be 5' or 3' depending on the option chosen in the first dialog. Note that for 3' trim, we internally reverse-complement the read and look for a match at the 5' end of the reverse complemented sequence. So in case of 3' trim, if a match is found at the 5' end, it will be treated as an internal match, because it is on the end of the sequence that is not being trimmed.

If a match can be treated as either an end match or an internal match, the workbench will treat it as an end match.

This section allows you to decide whether to

- Allow internal matches

- Allow end matches
- Allow both internal and end matches

You can also change the minimum scores for both internal and end score

- Minimum internal score is set to 10 by default
- Minimum end score is set to 4 by default

End matches have usually a lower score, as adapters found at the end of reads may be incomplete.

For example, if your adapter is 8 nucleotides long, it will never be found in an internal position with the settings set as they are by default (the minimum internal score being at 10).

Figure 21.8 shows a few examples with an adapter match at the end.

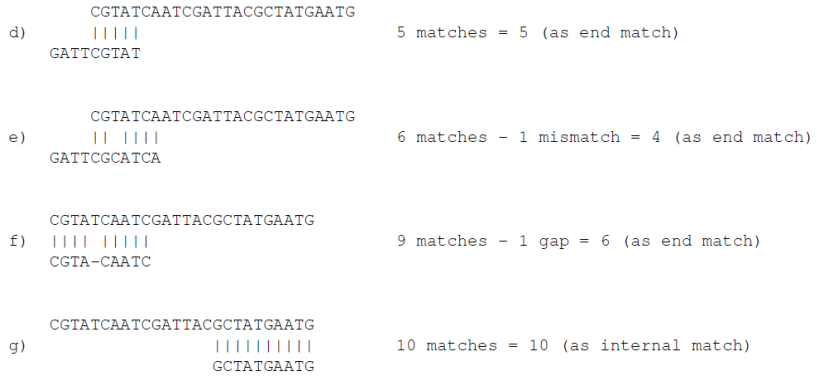


Figure 21.8: *Four examples showing a sequencing read (top) and an adapter (bottom). The examples are artificial.*

In the first two examples (d and e), the adapter sequence extends beyond the end of the read. This is what typically happens when sequencing small RNAs where you sequence part of the adapter. The third example (f) shows a case that could be interpreted both as an end match and an internal match. However, the workbench will interpret this as an end match, because it starts at beginning (5' end) of the read. Thus, the definition of an end match is that the alignment of the adapter starts at the read's 5' end. The last example (g) could also be interpreted as an end match, but because it is a the 3' end of the read, it counts as an internal match (this is because you would not typically expect partial adapters at the 3' end of a read).

Below (figure 21.9), the same examples are re-iterated showing the results when applying different scoring schemes. In the first round, the settings are:

- When an adapter is found: Remove adapter and the preceding sequence (5' trim)
- Allowing internal matches with a minimum score of 6
- Not allowing end matches

A different set of adapter settings could be:

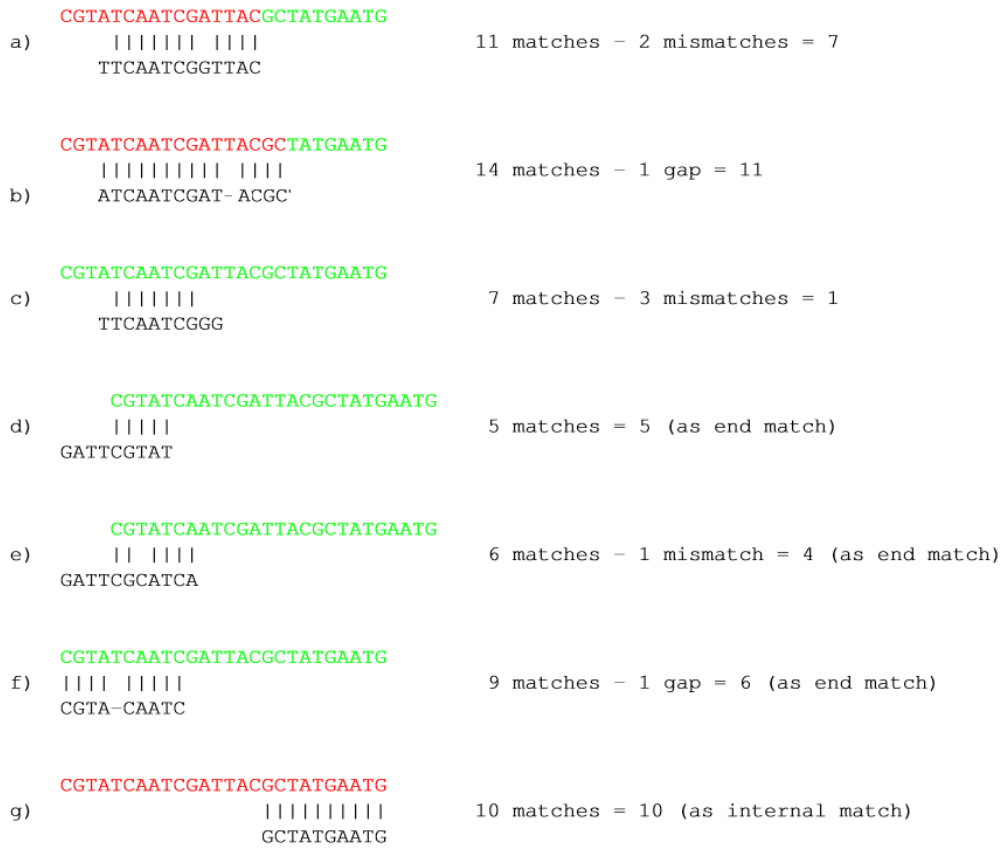


Figure 21.9: *The results of trimming with internal matches only. Red is the part that is removed and green is the retained part. Note that the read at the bottom is completely discarded.*

- When an adapter is found: Remove adapter and the preceding sequence (5' trim)
- Allowing internal matches with a minimum score of 11
- Allowing end match with a minimum score of 4

The results of such settings is shown in figure 21.10.

Click on the button labeled **Finish** to create the trim adapter list. You must now save the generated trim adapter list in the **Navigation Area**. You can do this by clicking on the tab and dragging and dropping the trim adapter list to the desired destination, or you can go to **File** in the menu bar and the choose **Save as**.

Creating a Trim adapter list based on the Trim Reads report

Since the automatic option works conservatively with data of low quality, you can benefit from creating a new Adapter Trim List based on the report generated by running the Trim Reads tool a first time.

1. Start the Trim Reads tool.
2. Select the reads you want to analyze (or a subset of these).

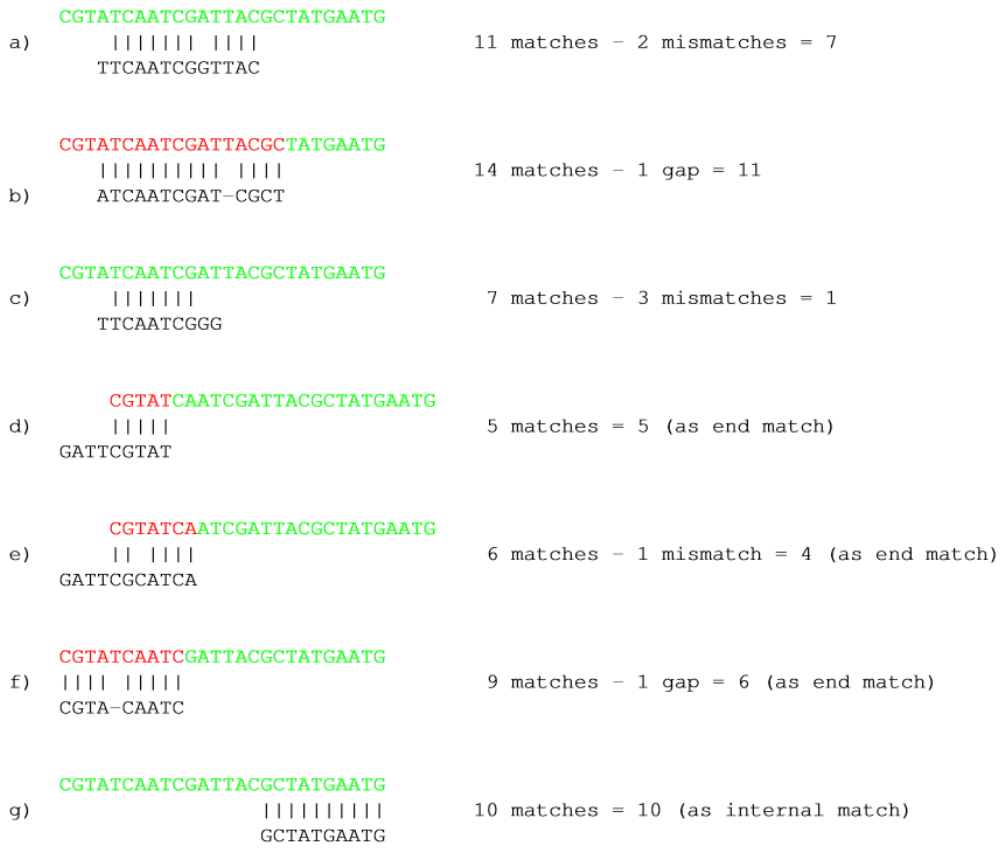


Figure 21.10: *The results of trimming with both internal and end matches. Red is the part that is removed and green is the retained part.*

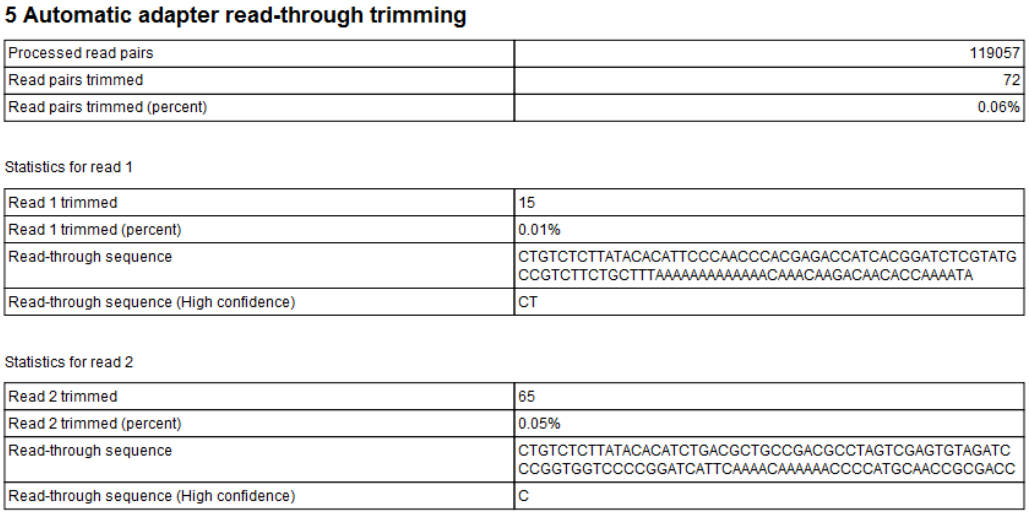
3. Leave the Quality trimming settings as they are set by default.
4. In the Adapter trimming step, make sure that the option "Automatic read-through adapter trimming" is selected and that no Adapter Trim List is specified.
5. Leave the Sequence filtering settings at their default value, i.e. with no filtering.
6. In the Result handling step ensure that "Create Report" is selected and click Finish.

Once the process is completed, open the report and scroll down to the last section named "5 Automatic adapter read-through trimming" (as seen in figure 21.11).

- If the detected "Read-through sequence" is < 10 bp, read-through adapters are not a big issue in your data and it can be trimmed using the "Automatic read-through adapter trimming" on its own. You do not need to re-run the tool with an adapter trimming list.
- If the detected "Read-through sequence" is > or equal to 10 bp, we recommend that you re-run the Trim Reads tool with a Trim adapter list generated using the report results.

To create a Trim adapter list with the read-though sequence from the report:

1. In the report, copy the sequence of the detected "Read-through sequence". If the sequence is long, then copy only the first 19 to 24 bp.



5 Automatic adapter read-through trimming

Processed read pairs	119057
Read pairs trimmed	72
Read pairs trimmed (percent)	0.06%

Statistics for read 1

Read 1 trimmed	15
Read 1 trimmed (percent)	0.01%
Read-through sequence	CTGTCTCTTATACACATTCCCAACCCACGAGACCATCACGGATCTCGTATG CCGTCTTCTGCTTTAAAAAAAAAAAACAAACAAGACAACACCAAAATA
Read-through sequence (High confidence)	CT

Statistics for read 2

Read 2 trimmed	65
Read 2 trimmed (percent)	0.05%
Read-through sequence	CTGTCTCTTATACACATCTGACGCTGCCGACGCCTAGTCGAGTGTAGATC CCGGTGGTCCCCGGATCATTCAAAACAAAAAACCCCATGCAACCGCGACC
Read-through sequence (High confidence)	C

Figure 21.11: *Use the statistics of the read-through trimming to create a Trim adapter list.*

2. Go to **New | Adapter Trim List**.
3. Click the **Add row** button.
4. Type the name of the first adapter, for example Read 1 read-through adapter.
5. Paste the copied sequence.
6. Set the Reads option to **First read**.
7. Choose the option **Remove the adapter and the following sequence (3' trim)**.
8. For reads without adapters choose the option **Keep the Read**.
9. In the Set scoring dialog, leave the default settings and click **Finish**.
10. Repeat for the procedure with the read-through sequence for read 2.
11. Save the Trim adapter list before closing it.

You can now use this Trim adapter list in combination with the "Automatic read-through adapter trimming" option for optimal adapter trimming of all samples in your experiment.

Importing an adapter list

It is possible to import a trim adapter list from an Excel or CSV file, using the Standard import with either the Automatic Import option, or the Force Import as Type: Trim Adapter List option.

To import a trim adapter list, the names of all adapters must be unique - the workbench is unable to accept files with multiple rows containing the same adapter name. The file must also include the following information: Name, Sequence, Reads, When an adapter is found, For reads without adapters, Alignment score

For CSV file, the text between each comma that designates a new column should be quoted, as shown in figure 21.12:

```
"Name","Sequence","Reads","When an adapter is found","For reads without adapters","Alignment score"
"ATC 5' trim all","ACGCTAGTCATA","All","Trim 5' end","Keep the read","Mismatch: 2, Gapcost: 3, Cutoff: 10, Cutoff at end: 4"
"ATC 3' trim first read","ACGCTAGTCAGTCTA","First read","Trim 3' end","Keep the read","Mismatch: 2, Gapcost: 3, Cutoff: 10, Cutoff at end: 4"
"ATC discard second read","ACGCTGTCAGTCAGTCTA","Second read","Trim 3' end","Keep the read","Mismatch: 2, Gapcost: 3, Cutoff: 10, Cutoff at end: 4"
"ATC discard reads when not found","ACGCTAGTCAGTCTA","All","Trim 5' end","Discard the read","Mismatch: 2, Gapcost: 3, Cutoff: 10, Cutoff at end: 4"
```

Figure 21.12: *The expected import format for Adapter Lists.*

21.2.4 Length trimming

Clicking **Next** will allow you to specify length trimming as shown in figure 21.13.

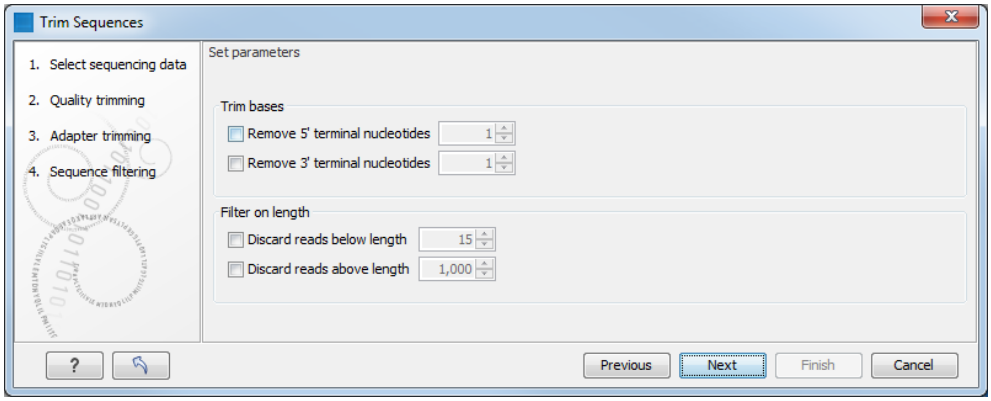


Figure 21.13: *Trimming on length.*

At the top you can choose to **Trim bases** by specifying a number of bases to be removed from either the 3' or the 5' end of the reads. Below you can choose to **Discard reads below length**. This can be used if you wish to simply discard reads because they are too short. Similarly, you can discard reads above a certain length. This will typically be useful when investigating e.g. small RNAs (note that this is an integral part of the small RNA analysis together with adapter trimming).

21.2.5 Trim output

Clicking **Next** will allow you to specify the output of the trimming as shown in figure 21.14.

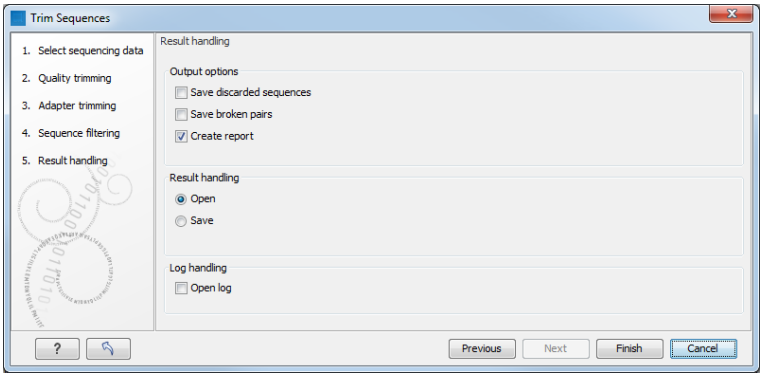


Figure 21.14: *Specifying the trim output.*

In most case, independently of what option are selected in this dialog, a list of trimmed reads will be generated:

- Sequence elements (individual sequences) selected as input and not discarded during trimming will be output into a single sequence list, as long as one or more of the input sequences were trimmed.

- Sequence lists selected as input will be output as as many corresponding sequence list, assuming that at least one sequence in any one of the sequence lists input was trimmed.

However, if no sequences are trimmed using the parameter settings provided, then no sequence lists are output when running the tool directly. A warning message appears stating that no sequences were trimmed. When the tool is run within a workflow, and if no sequences are trimmed using the parameter settings provided, then all input sequences are passed to the next step of the analysis via the "Trimmed Sequences" output channel.

In addition the following can be output as well:

- **Save discarded sequences**. This will produce a list of reads that have been discarded during trimming. Sections trimmed from reads that are not themselves discarded will not appear in this list.
- **Save broken pairs**. This will produce a list of orphan reads.
- **Create report**. An example of a trim report is shown in figure 21.15. The report includes the following:
  - **Trim summary.**
    - **Name.** The name of the sequence list used as input.
    - **Number of reads.** Number of reads in the input file.
    - **Avg. length.** Average length of the reads in the input file.
    - **Number of reads after trim.** The number of reads retained after trimming. This includes both paired and orphan reads.
    - **Percentage trimmed.** The percentage of the input reads that are retained.
    - **Avg. length after trim.** The average length of the retained sequences.
  - **Read length before / after trimming**. This is a graph showing the number of reads of various lengths. The numbers before and after are overlayed so that you can easily see how the trimming has affected the read lengths (right-click the graph to open it in a new view).
  - **Trim settings** A summary of the settings used for trimming.
  - **Detailed trim results**. A table with one row for each type of trimming:
    - **Input reads.** The number of reads used as input. Since the trimming is done sequentially, the number of retained reads from the first type of trim is also the number of input reads for the next type of trimming.
    - **No trim.** The number of reads that have been retained, unaffected by the trimming.
    - **Trimmed.** The number of reads that have been partly trimmed. This number plus the number from **No trim** is the total number of retained reads.
    - **Nothing left or discarded.** The number of reads that have been discarded either because the full read was trimmed off or because they did not pass the length trim (e.g. too short) or adapter trim (e.g. if **Discard when not found** was chosen for the adapter trimming).
  - **Automatic adapter read-through trimming**. This section contains statistics about how many reads were automatically trimmed for adapter read-through. It will also list the two detected read-through sequences.

1 Trim summary

Name	Number of reads	Avg.length	Number of reads after trim	Percentage trimmed	Avg.length after trim
reads	57.213	228,0	55.754	~100%	232,8

2 Read length before / after trimming

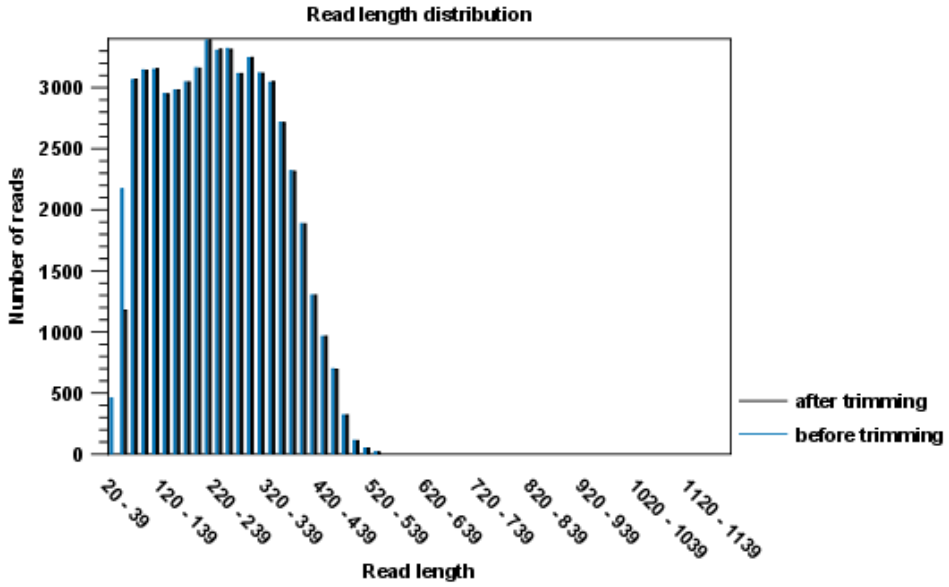


Figure 21.15: *A report with statistics on the trim results. Note that the Average length after trimming (232,8bp) is bigger than before trimming (228bp) because 2.000 very short reads were discarded in the trimming process.*

If you trim paired data, the result will be a bit special. In the case where one part of a paired read has been trimmed off completely, you no longer have a valid paired read in your sequence list. In order to use paired information when doing assembly and mapping, the Workbench therefore creates two separate sequence lists: one for the pairs that are intact, and one for the single reads where one part of the pair has been deleted. When running assembly and mapping, simply select both of these sequence lists as input, and the Workbench will automatically recognize that one has paired reads and the other has single reads.

21.3 Demultiplex reads

Multiplexing techniques are often used when sequencing different samples in one sequencing run. One method used is to *tag* the sequences with a unique identifier during the preparation of the sample for sequencing [Meyer et al., 2007].

With this technique, each sequence read will have a sample-specific tag, which is a specific sequence of nucleotides before and after the sequence of interest. This principle is shown in figure 21.16.

The sample-specific tag, also called the barcode or the index, can then be used to distinguish between the different samples when analyzing the sequencing data.

Figure 21.16: *Tagging the target sequence, which in this case is single reads from one sample.*

Post-processing of the sequencing data is required to separate the reads into their corresponding samples. Based on their barcodes this can be done using the demultiplexing functionality of the *Biomedical Genomics Workbench*. Using this tool, sequences are associated with a particular sample when they contain an exact match to a particular barcode. Sequences that do not contain an exact match to any of the barcode sequences provided are classified as not grouped and are put into a sequence list with the name "Not grouped".

Note that there is also an example using Illumina data at the end of this section.

Before processing the data, you need to import it as described in section 6.3.

Please note that demultiplexing is often carried out on the sequencing machine so that the sequencing reads are already separated according to sample. This is often the best option, if it is available to you. Of course, in such cases, the data will not need to be demuliplexed again after import into the *Biomedical Genomics Workbench*.

To demultiplex your data, please go to:

Toolbox | **Preparing Raw Data (**  **)** | **Demultiplex Reads (**  **)**

This opens a dialog where you can specify the sequences to process (figure 21.17).

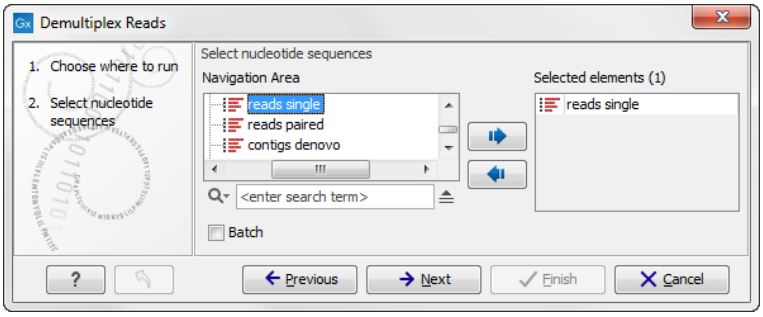


Figure 21.17: *Specify the sequences to demultiplex.*

When you click on the button labeled **Next**, you can then specify the details of how the demultiplexing should be performed. At the bottom of the dialog, there are three buttons, which are used to **Add**, **Edit**, and **Delete** the elements that describe how the barcode is embedded in the sequences.

First, click **Add** to define the first element. This will bring up the dialog shown in 21.18.

At the top of the dialog, you can choose the type of element you wish to define:

- **Linker**. The linker (also known as adapter) is a sequence which should just be ignored - it is neither the barcode nor the sequence of interest. In the example in figure 21.16, the linker is two nucleotides long. For this element, you simply define its length - nothing else.
- **Barcode**. The barcode (also known as index) is the stretch of nucleotides used to group

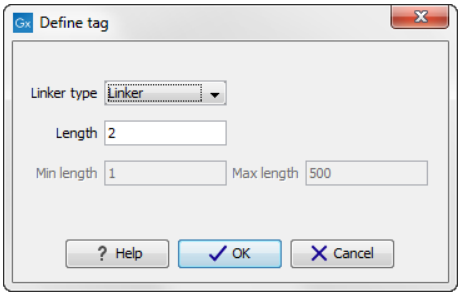


Figure 21.18: *Defining an element of the barcode system.*

the sequences. In this dialog, you simply need to specify the length of the barcode. The valid sequences for your barcodes must be provided at a later wizard step.

- **Sequence**. This element defines the sequence of interest. You can define a length interval for how long you expect this sequence to be. The sequence part is the only part of the read that is retained in the output. Both barcodes and linkers are removed.

The concept when adding elements is that you add e.g. a linker, a barcode, and a sequence in the desired sequential order to describe the structure of each sequencing read. You can of course edit and delete elements by selecting them and clicking the buttons below. In the example shown in figure 21.16, the dialog should include a linker, a barcode, and a sequence as shown in figure 21.19.

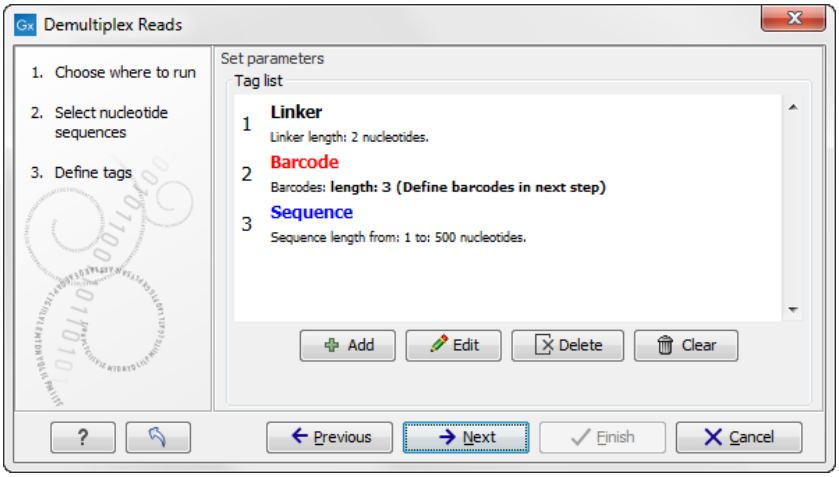


Figure 21.19: *Processing the tags as shown in the example of figure 21.16.*

Click on the button labeled **Next** to go to the last wizard step, where you can specify the output options. If you choose to keep the default settings, three different types of output will be generated; 1) The demultiplexed reads, one output for each specified barcode (the file name starts with "Barcode:" and is followed by the specified barcode sequence), 2) The discarded reads that did not have a barcode (this file is called "Not grouped"), and 3) a "Demultiplex Reads report", which shows the fraction of reads with and without a barcode (see figure 21.20).

If you have paired data the procedure is exactly the same, except for one thing: the dialog shown in figure 21.19 will be displayed twice - one for each part of the pair.

Figure 21.21 shows an example that could illustrate paired end reads. In this example we have paired end reads from two different samples mixed together.

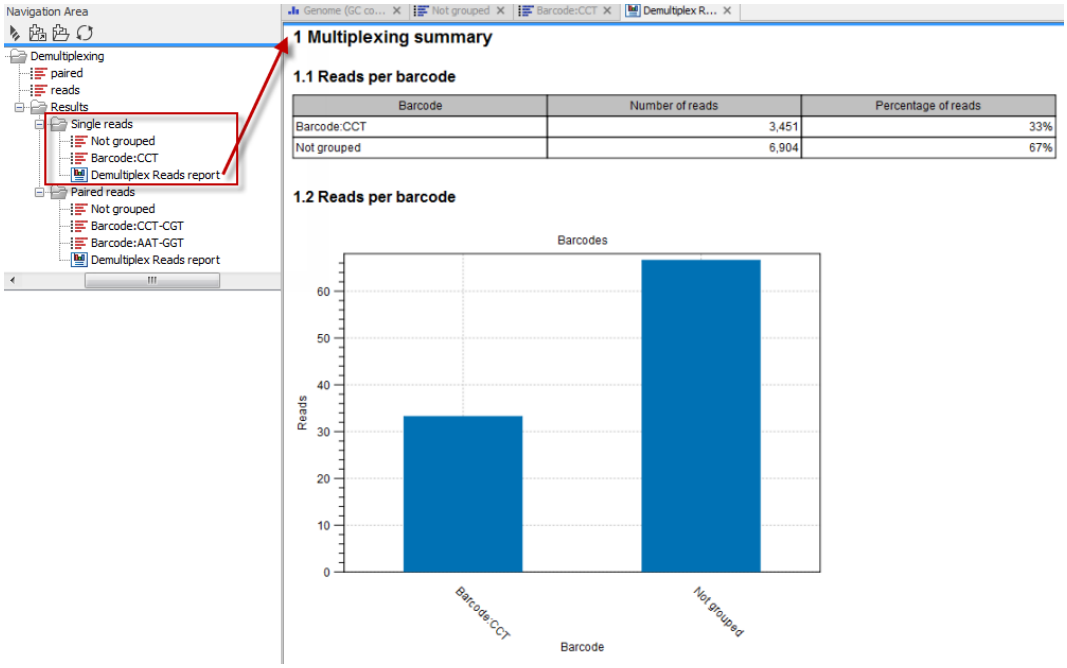


Figure 21.20: *Three different outputs are generated when analyzing single reads with only one sample using the default output settings. If several samples had been mixed together there would be a sequence list for each sample (each specified barcode). The Demultiplex Reads report is shown in the right-hand side of the figure.*

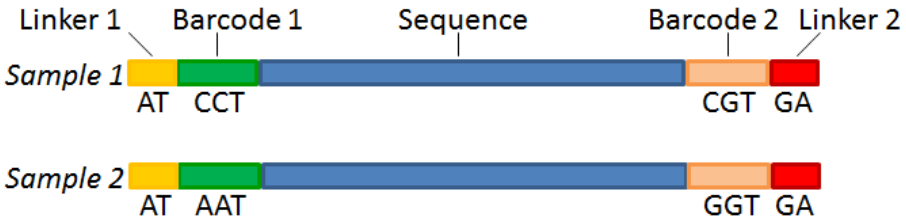


Figure 21.21: *Paired end reads with linkers and barcodes. Two different samples are mixed together in this example.*

In a situation where the paired reads are expected to be barcoded in the same way (see example below), you would set the parameters for read1 (wizard step 3) and read2 (wizard step 4) to be the same.

Read1 : –Linker–Barcode1–Sequence

Read2 : –Linker–Barcode1–Sequence

However, if read2 of the pair is not expected to be the same as read1 in the pair, it is necessary to adjust these settings accordingly. For example, it is possible that read2 does not contain any barcode sequence at all. In this case, you would simply set the sequence parameter for the mate and exclude the barcode and linker parameters. If the two reads in the read pair have different barcodes, the situation would look like this:

Read1 : –Linker–Barcode1–Sequence

Read2 : –Linker–Barcode2–Sequence

To demultiplex paired end reads the first two steps are similar to the demultiplexing of single reads. Select the paired end read sequences that should be demultiplexed (figure 21.22).

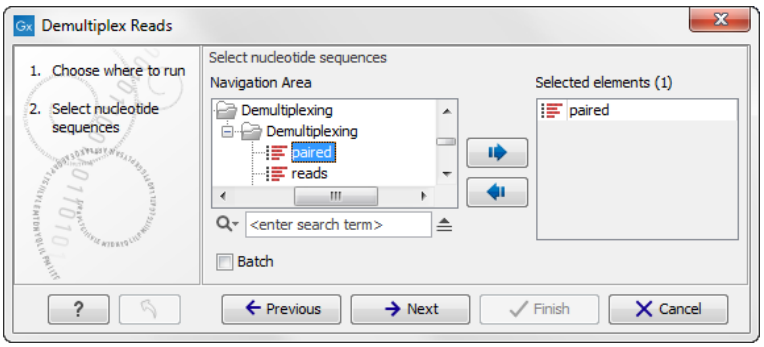


Figure 21.22: *Specifying the paired end reads to demultiplex.*

Click on the button labeled **Next** to define the tags and sequence for the forward reads (figure 21.23).

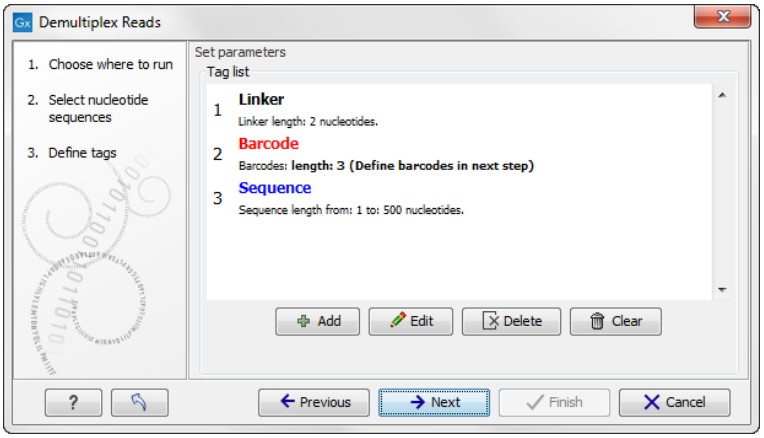


Figure 21.23: *Specifying the setup of the forward read (tags and sequence).*

Barcodes can be entered manually or imported from a properly formatted CSV or Excel file:

Manually The barcodes can be entered manually by clicking the **Add** ( ) button. In the example shown in figure 21.21 the barcodes should be defined as shown in figure 21.24.

You can edit the barcodes and the names by clicking the cells in the table. The barcode name is used when naming the results.

Import from CSV or Excel To import a file of barcodes, click on the **Import** ( ) button. The input format consists of two columns: the first contains the barcode sequence, the second contains the name of the barcode. An acceptable csv format file would contain columns of information that looks like:

```
 "AAAAAA","Sample1"
"GGGGGG","Sample2"
 "CCCCCC","Sample3"
```

Note that double quotes around values are always necessary regardless of whether columns are separated with a comma or a semi-colon.

The **Preview** column will show a preview of the results by running through the first 10,000 reads

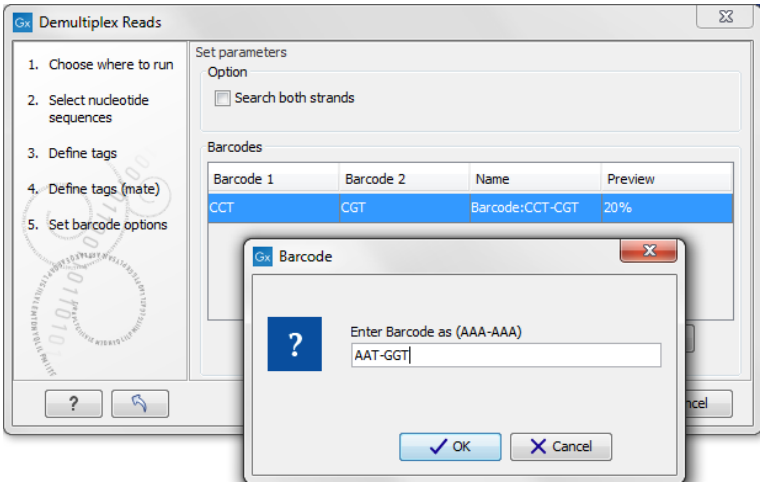


Figure 21.24: *The barcodes for the set of paired end reads for sample 1 have already been defined and the barcodes for sample 2 is being entered in the format AAA-AAA, which corresponds to Barcode1-Barcode2 for sample 2 in the example shown in figure 21.21.*

(figure 21.25).

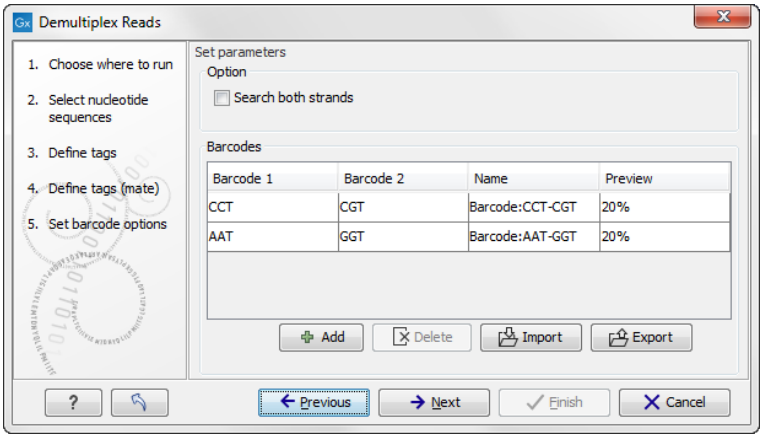


Figure 21.25: *A preview of the results.*

At the top, you can choose to search on both strands for the barcodes (this is needed for some 454 protocols where the MID is located at either end of the read).

If you would like to change the name of the sequence(s), this can be done at this step by double-clicking on the specific name that you would like to change. This is shown in figure 21.26.

Click on the button labeled **Next** to define the tags and sequence for the reverse reads (figure 21.27).

Click **Next** to specify the output options. First, you can choose to create a list of the reads that could not be grouped. Second, you can create a summary report showing how many reads were found for each barcode. Click **Finish** to perform the demultiplexing.

There is also an option to create subfolders for each sequence list. This can be handy when the results need to be processed in batch mode (see section 8.3).

A new sequence list will be generated for each barcode containing all the sequences where this barcode is identified. Both the linker and barcode sequences are removed from each of

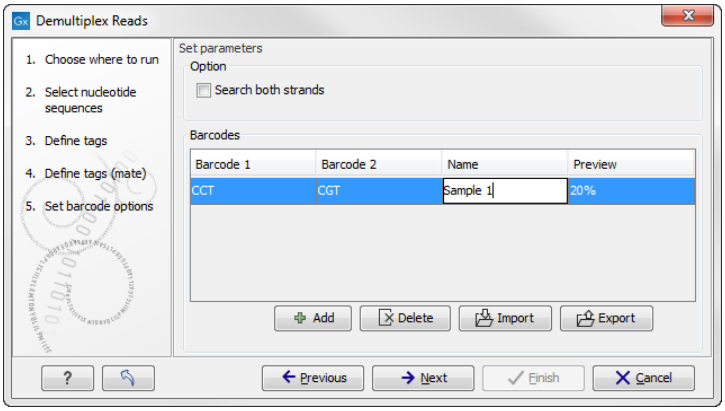


Figure 21.26: *The name of the sequence can be renamed by double-clicking on the existing name.*

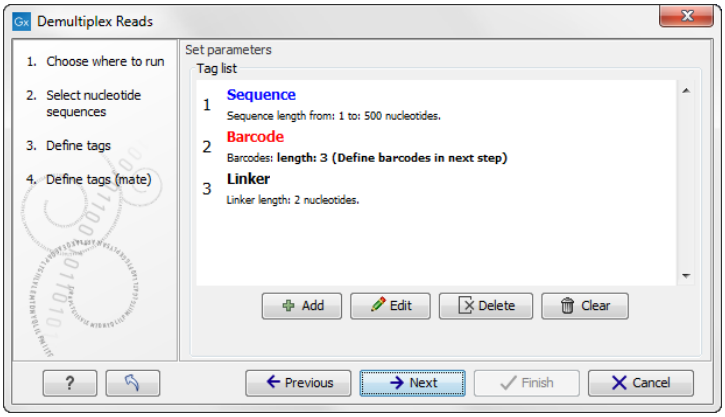
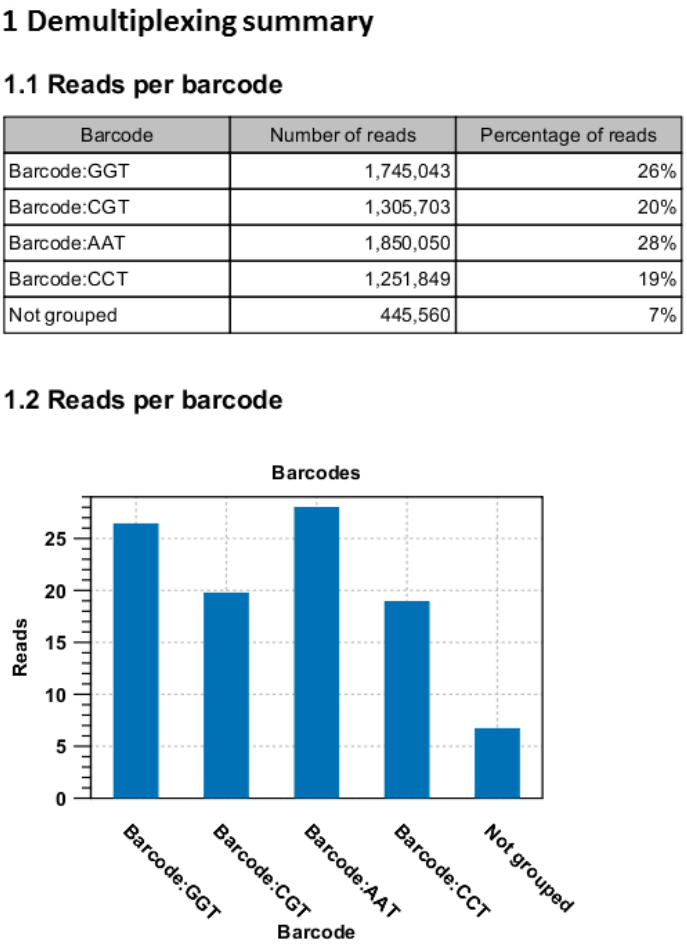


Figure 21.27: *Specifying the setup of the reverse read (tags and sequence).*

the sequences in the list, so that only the target sequence remains. This means that you can continue the analysis by doing trimming or mapping. Note that you have to perform separate mappings for each sequence list.

An example of the demultiplexing summary report is shown in figure 21.28.



1 Demultiplexing summary

1.1 Reads per barcode

Barcode	Number of reads	Percentage of reads
Barcode:GGT	1,745,043	26%
Barcode:CGT	1,305,703	20%
Barcode:AAT	1,850,050	28%
Barcode:CCT	1,251,849	19%
Not grouped	445,560	7%

1.2 Reads per barcode

Figure 21.28: *An example of a report showing the number of reads in each group. In this example four different barcodes were used to separate four different samples.*

21.3.1 An example using Illumina barcoded sequences

The data set in this example can be found at the Short Read Archive at NCBI. Use the Search for Reads in SRA... tool to search for SRX014012. Select the SRR03730 item and click **Download Reads and Metadata**. Save the sequence list in the Navigation Area, and use it with the Demultiplex Reads tool.

The barcoding was done using the following tags at the beginning of each read: CCT, AAT, GGT, CGT (see supplementary material of Cronn et al., 2008). The settings in the dialog should thus be as shown in figure 21.29.

Click next to the "Set barcode options" dialog and use the **Add** button) to specify the bar codes as shown in figure 21.30.

With this data set we got the four groups as expected (shown in figure 21.31). The **Not grouped** list contains 445,560 reads that will have to be discarded since they do not have any of the barcodes.

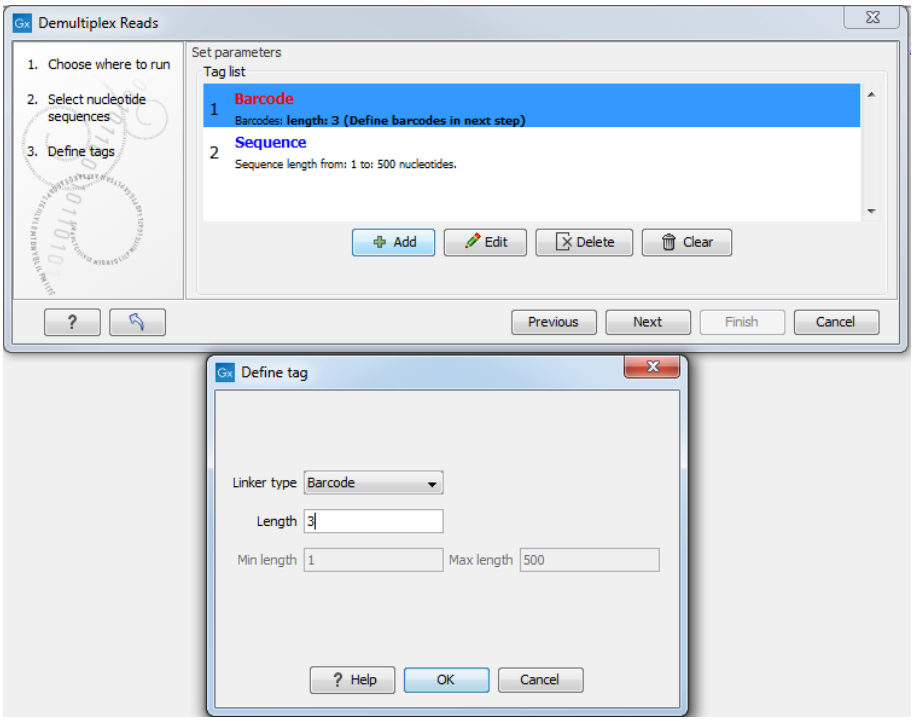


Figure 21.29: *Setting the barcode length at three.*

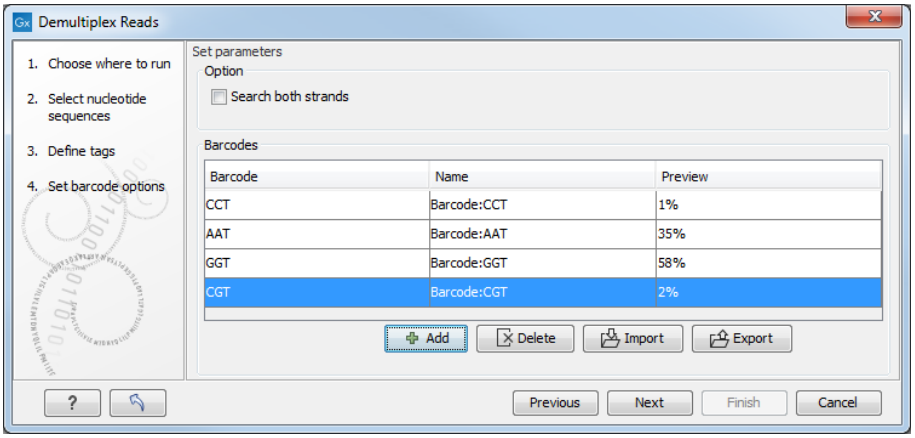


Figure 21.30: *A preview of the result*

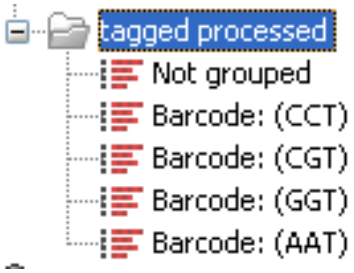


Figure 21.31: *The result is one sequence list per barcode and a list with the remainders*

Part VI

Resequencing analysis

Chapter 22

Resequencing analysis tools

Contents

In the *Biomedical Genomics Workbench resequencing* is the overall category for applications comparing genetic variation of a sample to a reference sequence. This can be targeted resequencing of a single locus or whole genome sequencing. The overall workflow will typically involve read mapping, some sort of variant detection and interpretation of the variants.

This chapter describes the tools relevant for the resequencing workflows downstream from the actual read mapping.

22.1 Map Reads to Reference

Read mapping is a very fundamental step in most applications of high-throughput sequencing data. The *Biomedical Genomics Workbench* includes read mapping in several other tools (such as in the Map Reads to Contigs tool, or for RNA-Seq Analysis), but this chapter will focus on the core read mapping algorithm. At the end of the chapter you can find descriptions of the read mapping reports and a tool to merge read mappings.

There are two different versions of the core mapper: one for color space data, and one for base space data. At http://www.qiagenbioinformatics.com/support/resources/ you can find white papers with detailed benchmarks and descriptions of both algorithms.

In addition, the mapper has been improved to work with PacBio reads and reads longer than 500bp. Before the Map Reads to Reference tool starts to map the reads, it checks the input sequence list(s) to decide on the mapping algorithm to use:

- If color space information is available, then the reads are mapped in color space. This is done using the legacy mapper (which is the standard mapper on the CLC Genomics Workbenches version 5.5 and earlier), which does not make use of the read group information.
- If no color space information is available, and the input sequence list(s) have the read group set to "PacBio", then the specialized mapping algorithm which is better suited for mapping long reads with many sequencing errors is applied.
- If no color space information is available, and the input sequence list(s)' read group is not set to "PacBio", then the reads are mapped using our standard mapping algorithm. The standard mapping algorithm uses the same seeding method for all input reads, but different extension methods for long (>500 bp) and short reads.

It is possible to mix sequence list that have the read group "PacBio" with sequence lists that have a different read group for the same mapping. In this case the appropriate mapping algorithm will be applied to each of the sequence list.

In contrast it is not possible to mix color space and base space data.

22.1.1 Selecting reads and reference

To start the read mapping:

Toolbox | **Resequencing Analysis ()** | **Map Reads to Reference ()**

In the first dialog, select the sequences or sequence lists containing the sequencing data (figure 22.1). Please note that reads longer than 100,000 bases are not supported.

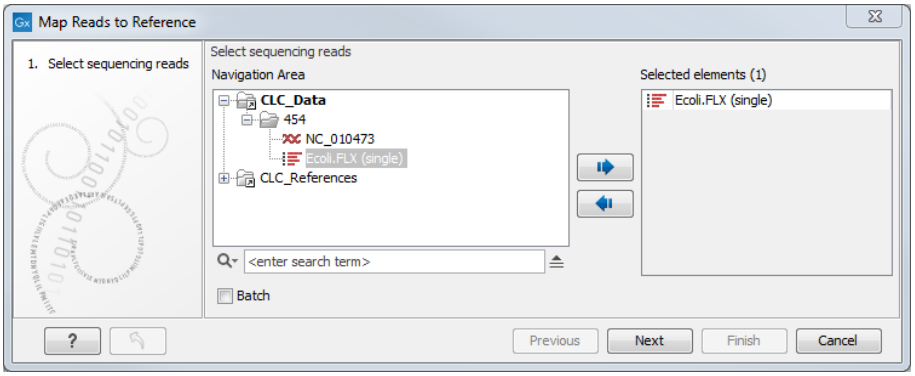


Figure 22.1: *Specifying the reads as input. You can also choose to work in batch.*

When the sequences are selected, click **Next**, and you will see the dialog shown in figure 22.2.

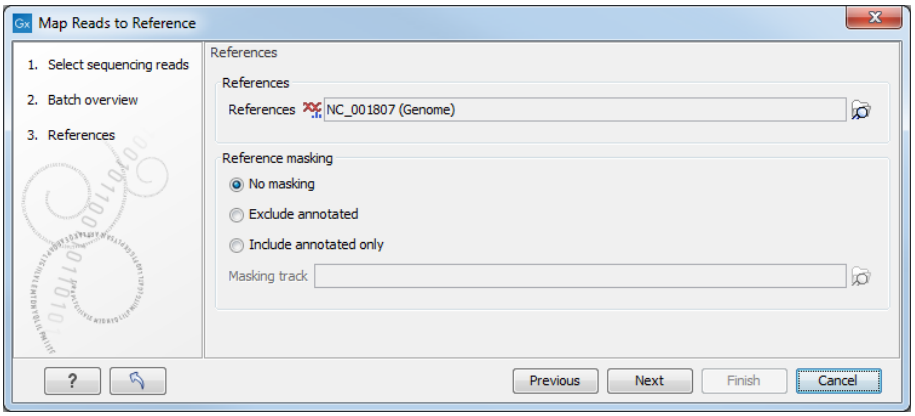


Figure 22.2: *Specifying the reference sequences and masking.*

At the top you select one or more reference sequences by clicking the **Browse and select element** ( ) button. You can select either single sequences, a list of sequences or a sequence track as reference. Note the following constraints:

- single reference sequences longer than 2gb ($2 \cdot 10^9$ bases) are not supported.
- a maximum of 120 input items (sequence lists or sequence elements) can be used as input to a single read mapping run.

22.1.2 Including or excluding regions (masking)

The next part of the dialog shown in figure 22.2 lets you *mask* the reference sequences. Masking refers to a mechanism where parts of the reference sequence are not considered in the mapping. This can be useful for example when mapping data is captured from specific regions (e.g. for amplicon resequencing). The read mapping will still base its output on the full reference - it is only the core read mapping that ignores regions.

Masking is performed by discarding the masked out nucleotides. As a result the reference is split into separate sequences, which are positioned according to the original unmasked reference sequence.

Note that you should be careful that your data is indeed only sequenced from the target regions. If not, some of the reads that would have matched a masked-out region perfectly may be placed wrongly at another position with a less-perfect match and lead to wrong results for subsequent variant calling. For resequencing purposes, we recommend testing whether masking is appropriate by running the same data set through two rounds of read mapping and variant calling: one with masking and one without. At the end, comparing the results will reveal if any off-target sequences cause problems in the variant calling.

Masking out repeats or using other masks with many regions is not recommended. Repeats are handled well and does not cause any slowdown. On the contrary, masking repeats is likely to cause a dramatic slowdown in speed, increase memory requirements and lead to incorrect read placement.

To mask a reference sequence, first click the **Include** or **Exclude** options, and second click the **Browse** ( ) button to select a track to use for masking.

22.1.3 Mapping parameters

Clicking **Next** leads to the parameters for the read mapping (see figure 22.3).

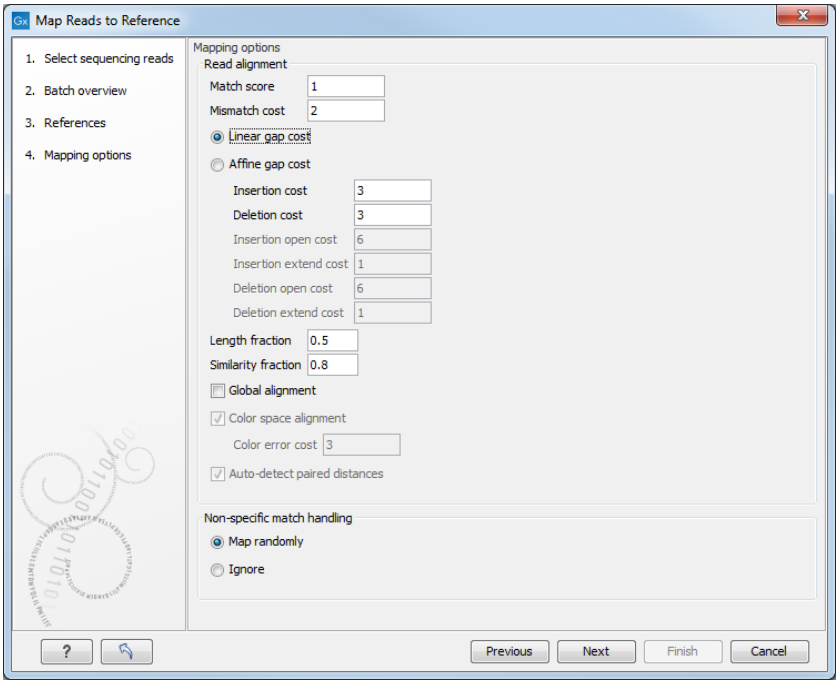


Figure 22.3: *Setting parameters for the mapping.*

The first parameter allows the mismatch cost to be adjusted:

- **Match score** The positive score for a match between the read and the reference sequence. It is set by default to 1 but can be adjusted up to 10.
- **Mismatch cost** The cost of a mismatch between the read and the reference sequence. Ambiguous nucleotides such as "N", "R" or "Y" in read or reference sequences are treated as a mismatches and any column with one of these symbols will therefore be penalized with the mismatch cost.

After setting the mismatch cost you need to choose between linear gap cost and affine gap cost, and depending on the model you chose, you need to set two different sets of parameters that control how gaps in the read mapping are penalized.

- **Linear gap cost** The cost of a gap is computed directly from the length of the gap and the insertion or deletion cost. This model often favors small, fragmented gaps over long contiguous gaps. If you choose linear gap cost, you must set the insertion cost and the deletion cost:

  Insertion cost The cost of an insertion in the read (a gap in the reference sequence). The cost of an insertion of length ℓ will be

  $$\ell \cdot \text{Insertion cost}, \tag{22.1}$$

  Deletion cost The cost of a deletion in the read (gap in the read sequence). The cost of a deletion of length ℓ will be

  $$\ell \cdot \text{Deletion cost}. \tag{22.2}$$

- **Affine gap cost** An extra cost associated with opening a gap is introduced such that long contiguous gaps are favored over short gaps. If you chose affine gap cost, you must also set the cost of opening an insertion or a deletion:

  Insertion open cost The cost of opening an insertion in the read (a gap in the reference sequence).

  Insertion extend cost The cost of extending an insertion in the read (a gap in the reference sequence) by one column.

  Deletion open cost The cost of a opening a deletion in the read (gap in the read sequence).

  Deletion extend cost The cost of extending a deletion in the read (gap in the read sequence) by one column.

  Using, affine gap cost, an insertion of length ℓ is penalized by

  $$\text{Insertion open cost} + \ell \cdot \text{Insertion extend cost}, \tag{22.3}$$

  and a deletion of length ℓ is penalized by

  $$\text{Deletion open cost} + \ell \cdot \text{Deletion extend cost}. \tag{22.4}$$

  In this way long consecutive gaps get a lower cost per column than small fragmented gaps and they are therefore favored.

The score of a match between the read and the reference is set to 1 by default. Adjusting the cost parameters above can improve the mapping quality, e.g. when the read error rate is high or the reference is expected to differ significantly from the sequenced organism. For example, if the reads are expected to contain many insertions and/or deletions, it can be a good idea to lower the insertion and deletion costs to allow more of such errors. However, one should also consider the possible drawbacks when adjusting these settings. For example, reducing the insertion and deletion cost increases the risk of mapping reads to the wrong positions in the reference.

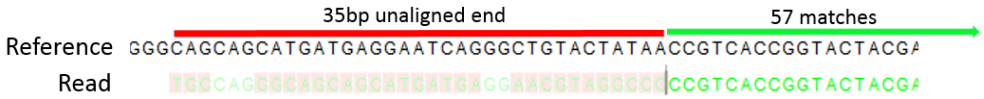


Figure 22.4: *An alignment of a read where a region of 35bp at the start of the read is unaligned while the remaining 57 nucleotides matches the reference.*

Figure 22.4 shows an example using linear gap cost where the read mapper is unable to map a region in a read due to insertions in the read and mismatches between the read and the reference. The aligned region of the read has a total of 57 matching nucleotides which result in an alignment score of 57 which is optimal when using the default cost for mismatches and insertions/deletions (2 and 3 respectively). If the mapper had aligned the remaining 35bp of the read as shown in Figure 22.5 using the default scoring scheme, the score would become:

$$(26 + 1 + 3 + 57) * 1 - 5 * 2 - 8 * 3 = 53 \tag{22.5}$$

In this case, the alignment shown in Figure 22.4 is optimal since it has the highest score. However, if either the cost of deletions or mismatches were reduced by one, the score of the alignment shown in Figure 22.5 would become 61 and 58, respectively, and thus make it optimal.

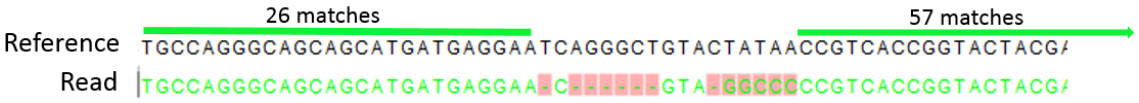


Figure 22.5: *An alignment of a read containing a region with several mismatches and deletions. By reducing the default cost of either mismatches or deletions the read mapper can make an alignment that spans the full length of the read.*

Once the optimal alignment of the read is found, based on the cost parameters described above, a filtering process determines whether this match is good enough for the read to be included in the output. The filtering threshold is determined by two factors:

- **Length fraction** The minimum percentage of the total alignment length that must match the reference sequence at the selected similarity fraction. A fraction of 0.5 means that at least half of the alignment must match the reference sequence before the read is included in the mapping (if the similarity fraction is set to 1). Note, that the minimal seed (word) size for read mapping is 15 bp, so reads shorter than this will not be mapped.
- **Similarity fraction** The minimum percentage identity between the aligned region of the read and the reference sequence. For example, if the identity should be at least 80% for the read to be included in the mapping, set this value to 0.8. Note that the similarity fraction relates to the length fraction, i.e. when the length fraction is set to 50% then *at least 50% of the alignment must have at least 80% identity* (see figure 22.6).
- **Global alignment** By default, mapping is done with **local alignment** of the reads to the reference. The advantage of performing local alignment instead of global alignment is that the ends are automatically left unaligned if there are many differences from the reference at the ends. For many sequencing platforms, the quality of the bases drop along the read, and a local alignment approach is desirable. Note that the aligned region has to be greater than the length threshold set. If **global alignment** is preferred, it can be enabled with a checkbox as shown in figure 22.3.

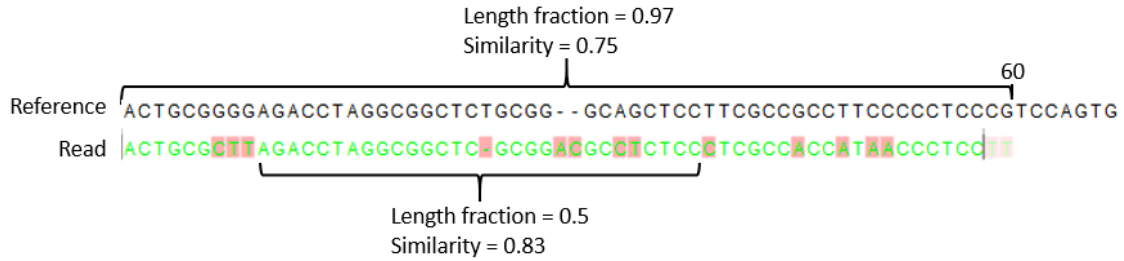


Figure 22.6: *A read containing 59 nucleotides where the total alignment length is 60. The part of the alignment that gave rise to the optimal score has length 58 which excludes 2 bases at the left end of the read. The length fraction of the matching region in this example is therefore 58/60 = 0.97. Given a minimum length fraction of 0.5, the similarity fraction of the alignment is computed as the maximum similarity fraction of any part of the alignment which constitute at least 50% of the total alignment. In this example the marked region in the alignment with length 30 (50% of the alignment length) has a similarity fraction of 0.83 which will satisfy the default minimum similarity fraction requirement of 0.8.*

- **Color space alignment** When mapping data in color space (data from SOLiD systems), the **color space** checkbox is enabled, and a corresponding cost for color errors can be set. If you do not have color space data, these will be disabled and are not relevant. For more details about this, please see section 22.4 which explains how color space mapping is performed in greater detail.

22.1.4 Mapping paired reads

- **Auto-detect paired distances** At the bottom of the dialog shown in figure 22.3 you can specify how **Paired reads** should be handled. You can read more about how paired data is imported and handled in section 6.3.7. If the sequence list used as input contains paired reads, this option will automatically be enabled - if it contains single reads, this option will not be applicable.

The *Biomedical Genomics Workbench* offers as the default choice to automatically calculate the distance between the pairs. If this is selected, the distance is estimated in the following way:

1. A sample of 100,000 reads is extracted randomly from the full data set and mapped against the reference using a very wide distance interval.
2. The distribution of distances between the paired reads is analyzed, and an appropriate distance interval is selected:
   - If less than 10,000 reads map, a simple calculation is used where the minimum distance is one standard deviation below the average distance, and the maximum distance is one standard deviation above the average distance.
   - If more than 10,000 reads map, a more sophisticated method is used which investigates the shape of the distribution and finds the boundaries of the peak.
3. The full sample is mapped using this distance interval.
4. The **history** ( ) of the result records the distance interval used.

The above procedure will be run for each sequence list used as input, assuming that they do not necessarily share the same library preparation and could have different distributions of paired distances. Figure 22.7 shows an example of the distribution of intervals with and without automatic pair distance interval estimation.

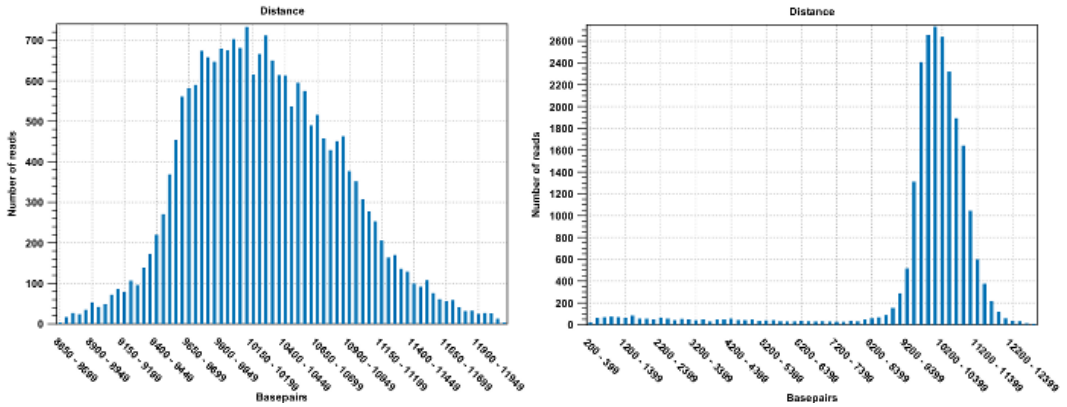


Figure 22.7: *To the left: mapping with a narrower distance interval estimated by the workbench. To the right: mapping with a large paired distance interval (note the large right tail of the distribution).*

Sometimes the automatic estimation of the distance between the pairs may return a warning "multiple intervals detected". This may happen if the reads derive from multiple libraries or from certain types of amplicon sequencing protocols. In this case, the estimates may still be correct, but, if in doubt, the user may want to disable the option to automatically estimate paired distances and instead manually specify minimum and maximum distances between pairs on the input sequence list.

If the automatic detection of paired distances is not checked, the mapper will use the information about minimum and maximum distance recorded on the input sequence lists (see section 6.3.7).

The 'automatic detection of paired distance' option when mapping should be used with caution. It is possible that the estimated distance setting is too narrow and consequently many read pairs will be flagged broken. Sometimes, a second peak in the Paired Distance distribution graph is not picked up on by the estimation tool.

If a large portion of pairs are flagged 'Broken' we recommend the following:

1. Inspect the detailed mapping report to deduce a distance setting interval - and compare this to the estimated distance used by the mapper (found in the mapping history).
2. Open the paired reads list and set a broad paired distance in the Elements tab. Then run a new mapping with the 'auto-detect...' OFF. Make sure to have a report produced. Open this report and look at the Paired Distance Distribution graph. This will tell you the distances that your pairs did map with. Use this information to narrow down the distance setting and perhaps run a third mapping using this.
3. Another cause of excessive amounts of broken pairs is misspecification of the read pair orientation. This can be changed in the Elements tab of the paired reads list prior to running a mapping.

See section 20.3 for further information about the mapping reports.

When a paired distance interval is set, the following approach is used for determining the placement of read pairs:

- First, all the optimal placements for the two individual reads are found.
- Then, the allowed placements according to the paired distance interval are found.
- If both reads can be placed independently but no pairs satisfies the paired criteria, the reads are treated as independent and marked as a **broken pair**.
- If only one pair of placements satisfy the criteria, the reads are placed accordingly and marked as uniquely placed even if either read may have multiple optimal placements.
- If several placements satisfy the paired criteria, the pair is treated as a non-specific match (see section 22.1.5 for more information.)
- If one read is uniquely mapped but the other read has several placements that are valid given the distance interval, the mapper chooses the location that is closest to the first read.

22.1.5 Non-specific matches

At the bottom of the dialog, you can specify how **Non-specific matches** should be treated. The concept of Non-specific matches refers to a situation where a read aligns at *more than one position with an equally good score*. In this case you have two options:

- **Random**. This will place the read in one of the positions randomly.
- **Ignore**. This will not include the read in the final mapping.

Note that a read is only considered non-specific when the read matches equally well at several alignment positions. If there are e.g. two possible alignment positions and one of them is a perfect match and the other involves a mismatch, the read is placed at the position with the perfect match and it is not marked as a non-specific match.

For paired data, reads are only considered non-specific matches if the entire pair could be mapped elsewhere with equal scores for both reads, or if the pair is broken in which case a read can be categorized as non-specific in the same way as single reads (see section 22.1.4).

When looking at the mapping, the default color for non-specific matches is yellow.

22.1.6 Gap placement

In the case of insertions or deletions in homopolymeric or repetitive regions, the precise placement of the insertion or deletion cannot be determined from the data. An example is shown in figure 22.8.

In this example, three A's in the reference (top) have been replaced by two A's in the reads (shown in red). The gap is placed towards the 5' end (left side), but could have been placed towards the 3' end with an equally good mapping score for the read as shown in figure 22.9.

Since either way of placing the gap is arbitrary, the goal of the mapper is to place the gaps consistently at the same side for all reads.

Figure 22.8: *Three A's in the reference (top) have been replaced by two A's in the reads (shown in red). The gap is placed towards the 5' end, but could have been placed towards the 3' end with an equally good mapping score for the read.*

Figure 22.9: *Three A's in the reference (top) have been replaced by two A's in the reads (shown in red). The gap is placed towards the 3' end, but could have been placed towards the 5' end with an equally good mapping score for the read.*

Many insertions and deletions in homopolymeric or repetitive regions reported in the public databases dbSNP and 1000Genomes have been identified based on mappings done with tools like BWA and Bowtie, that place insertions or deletions at the left side of a homopolymeric tract. Thus, to help facilitate the comparison of variant results with such public resources, the Map Reads to Reference tool will also place insertions or deletions in homopolymeric tracts at the left hand side.

For users of the COSMIC database or other clinical databases following the recommendations from the Human Genome Variation Society (HGVS) The Human Genome Variation Society (HGVS) recommendations, which pertain to variants within genes, state that for insertions and deletions in homopolymeric or repetitive regions, the most 3' position (corresponding to the strand of the gene) possible should be arbitrarily assigned as the site of change (see `http://varnomen.hgvs.org/`). Resources such as COSMIC adhere to these recommendations. In this case, placement to the farthest possible left hand position, as viewed in the *Biomedical Genomics Workbench*, of insertions or deletions in repetitive or homopolymeric tracts, has a different effect, depending on whether the gene involved is on the positive or negative strand of the reference. Such variants located within genes on the negative strand can be compared with the COSMIC database, while those within genes lying on the positive strand cannot be, as the positions relative to the reference will be different in this case.

22.1.7 Mapping computational requirements

The memory requirements of **Map Reads to Reference** depends on four factors. The size of the reference, the length of the reads, the read error rate and the number of CPU cores available. The limiting factor is often the size of the reference while the contribution of the other three

factors to the total memory consumption is usually small (see below).

A good estimate for the memory required by the base space read mapper to represent a reference is one MB for each Mbp in the reference. For example the human reference genome requires $3200 * 1MB = 3.2GB$ of memory. The color space mapper is able to scale down its memory consumption, such that even large references can be represented using small amounts of memory. However, when the memory consumption is scaled down it causes the read mapping to become slower.

When mapping short high quality reads, such as Illumina reads, the added memory consumption per CPU core is small. However, when mapping long reads with a high error rate, such as PacBio reads, each CPU core can add several hundred MB to the total memory consumption. Consequently, mapping long reads with high error rate on a machine with many CPU cores, can cause a large increase in the memory requirements for all CLC read mappers. An additional 4GB of memory should be reserved for the *Biomedical Genomics Workbench*, and thus the recommended minimum amount of memory for mapping short high quality reads (e.g. Illumina reads) to the human genome is 8GB.

22.1.8 Reference caching

In some cases repeated mappings against the same reference will result in a dramatically reduced runtime because the internal data structure used for mapping the reads, which is reference specific, can be reused. This has been enabled by storing files in the system tmp folder as a caching mechanism. Only a certain amount of disk space will be used and once reaching the limit, the oldest files are cleaned up. Consequently, the reference data structure files will automatically have to be recreated if the cache was filled or the tmp folder was cleaned up.

The default space limit is 8 GB which can be changed by going to

Edit | **Preferences** | **Advanced** | **Read Mapper**

On the server and for webstart the cache size can be controlled by creating a settings file "readmapper.properties" with an entry going like this "referencecachesize = 8589934592" where the size is in bytes. On grid setups, the "readmapper.properties" file will have to be added manually to each grid worker directory.

22.2 Mapping output

22.2.1 Mapping output options

Click **Next** lets you choose how the output of the mapping should be reported (see figure 22.10).

There are two independent output options available that can be (de-)activated in both cases:

- **Create report**. This will generate a summary report as described in section 22.3.
- **Collect un-mapped reads**. This will collect all the reads that could not be mapped to the reference into a sequence list (there will be one list of unmapped reads per sample, and for paired reads, there will be one list for intact pairs and one for single reads where the mate could be mapped).

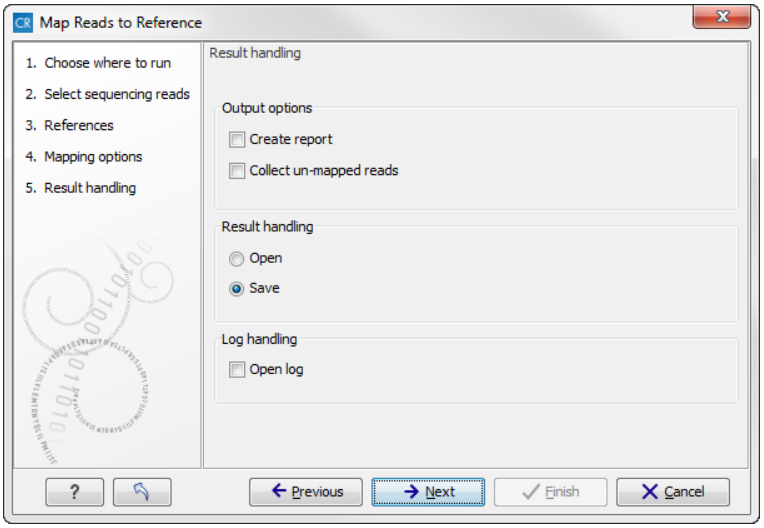


Figure 22.10: *Mapping output options.*

However, the main output is a reads track:

Reads track A reads track is very "lean" (i.e. with respect to memory requirements) since it only contains the reads themselves. Additional information about the reference, consensus sequence or annotations can be added and viewed alongside in the context of a Track List later (by adding, for example, a reference and/or annotation track, respectively). This kind of output is useful when working with tracks in general and especially for resequencing purposes this is recommended. Details about viewing and editing of reads-tracks are described in chapter 19.

Note that the Map Reads to Reference tool will output an empty read mapping and report when nothing mapped, and empty unmapped reads if everything mapped.

Finally, you can choose to save or open the results, and if you wish to see a log of the process, see section 8.2.

Clicking **Finish** will start the mapping.

22.2.2 Mapped reads coloring

The mapped reads are colored by default according to the following color code:

- Single reads mapping in their forward direction are green.
- Single reads mapping in their reverse direction are red.
- Paired reads are blue. Reverse paired reads are light blue (but only if the option "Highlight reverse paired reads" is checked, as it is by default, in read tracks). The thick line represents the read itself; the thin line represents the distance between each read in the pair. Note that reads from broken pairs are colored according to their forward/reverse orientation or as a non-specific match.
- Non-specific matches are yellow. When a read would have matched equally well another place in the mapping, it is considered a non-specific match. This color will "overrule" the

other colors. Note that when mapping to several reference sequences, i.e. chromosomes, a read is considered a double match when it matches more than once across all the chromosomes.

Unaligned ends, that is the part of the reads that is not mapped to the reference (also known as soft-clipped read ends) will be shown with a faded color, e.g. light green, light red, light blue or light yellow, depending on the color of the read.

Mismatches between the read and reference are shown as narrow vertical traits following the Rasmol color scheme (figure 22.11):

- A - red.
- T - green.
- C - blue.
- G - yellow.

Figure 22.11: *Mismatches between the reads and reference are shown as narrow vertical traits following the Rasmol color scheme.*

When zooming in at the nucleotide level, any mismatches between the read and reference are shown in black with a background color highlighting according to the Rasmol color scheme (see figure 22.12).

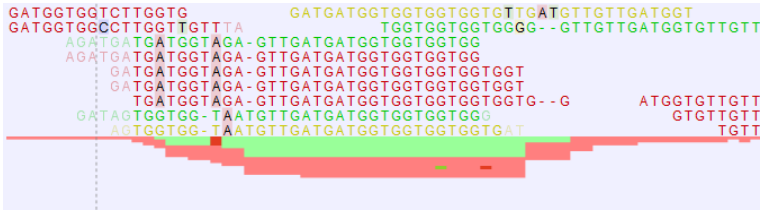


Figure 22.12: *At the nucleotide level, mismatches between the reads and reference are shown in black with a Rasmol background color. The overflow graph below the reads shows how many more forward or reverse reads cannot be displayed in the track. Expand the tracks to see all reads at the nucleotide level.*

At this level, usually more reads are mapped to the reference than can be shown in the track. In such cases the reads will be displayed in an overflow graph below the reads. The overflow graph is shown in the same colors as the sequences, and mismatches in reads are shown as horizontal lines following the Rasmol color scheme.

If your read track shows the message 'Too much data for rendering' on a grey background, simply zoom in to see your reads in more detail. This occurs when there are too many reads to be displayed clearly (more specifically when there are more than 500,000 reads displayed in the track, with paired reads counting as one.)

22.2.3 Reads track output from a read mapping

The side-panel functionality for viewing a read track and an annotation track is shown in Figure 22.17.

An aggregated Reads Track is shown in figure 22.13. The three blue shades shown in the Reads Track represent the aggregated mapped reads. This is displayed when the amount of data in view reaches the data aggregation setting. Within the specified aggregation bucket, from top to bottom the blue colors mean:

- The maximum coverage value (read count).
- The average coverage value.
- The minimum coverage value.

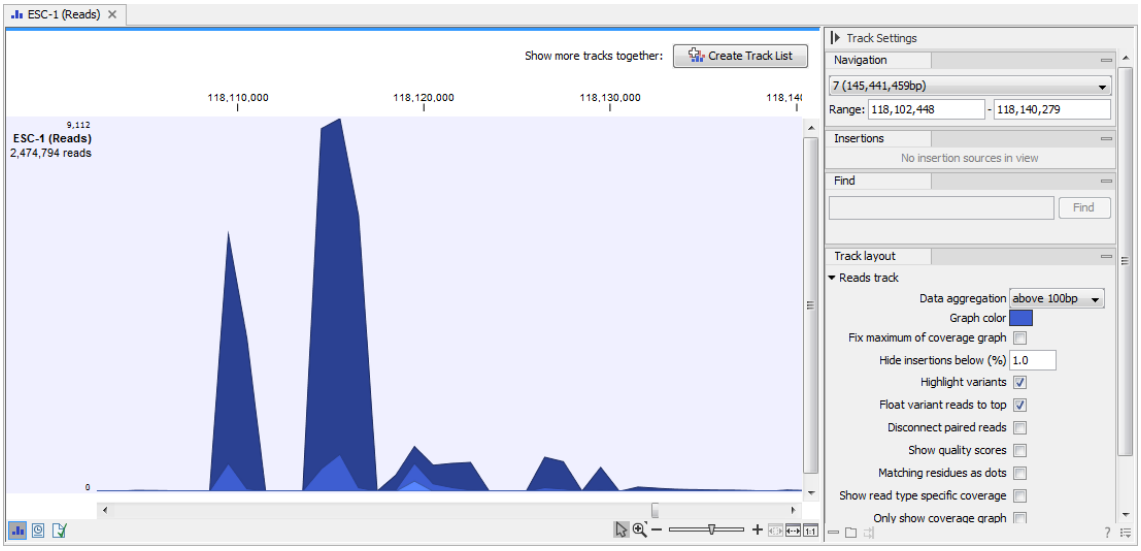


Figure 22.13: *In the aggregated Reads Track, the three blue shades represent the aggregated mapped reads.*

The data aggregation view is an alternative way of displaying a large amount of data (mapped reads) on the screen in order to shorten the data display time. This aggregated view allows you to navigate the view more smoothly and when used in combination with other tracks, this view can be used to get an overview of e.g. how many SNPs are located in a certain region.

In figure 22.14 we have zoomed in on a reads track (at the top showing the individual reads), with CDS and SNP annotations shown below.

If you zoom in further the alignment of the reads and the reference sequence can be viewed at single nucleotide level (see figure 22.15).

In this case only three reads are visible. In order to see more reads, increase the height of the reads track by dragging down the lower part of the track with the mouse (Figure 22.16).

The options for the **Side Panel** vary depending on which track is shown in the View Area. In figure 22.17 an example is shown for a read mapping:

Navigation. Gives information about which chromosome is currently shown. Below this, you can see the start and end positions of the shown region of the chromosome. The drop-down

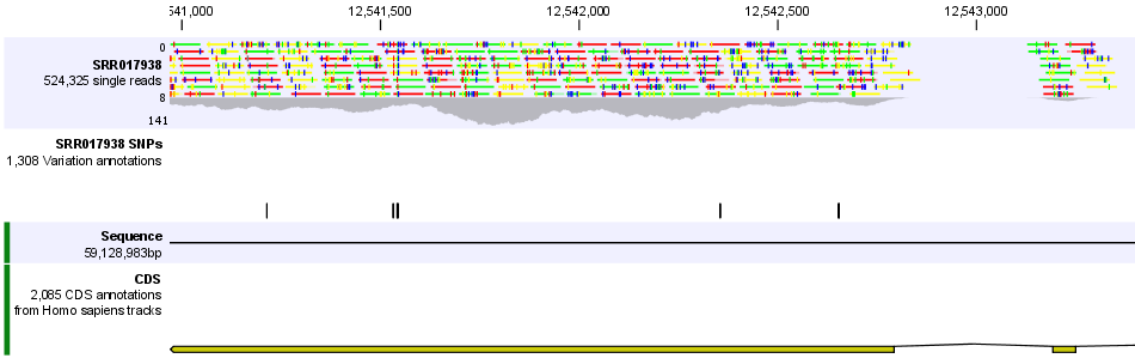


Figure 22.14: *Zooming in on the tracks reveals details.*

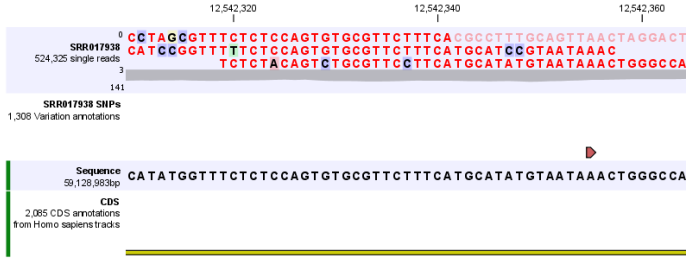


Figure 22.15: *Zoom in to see the bases of the reads and the reference sequence.*

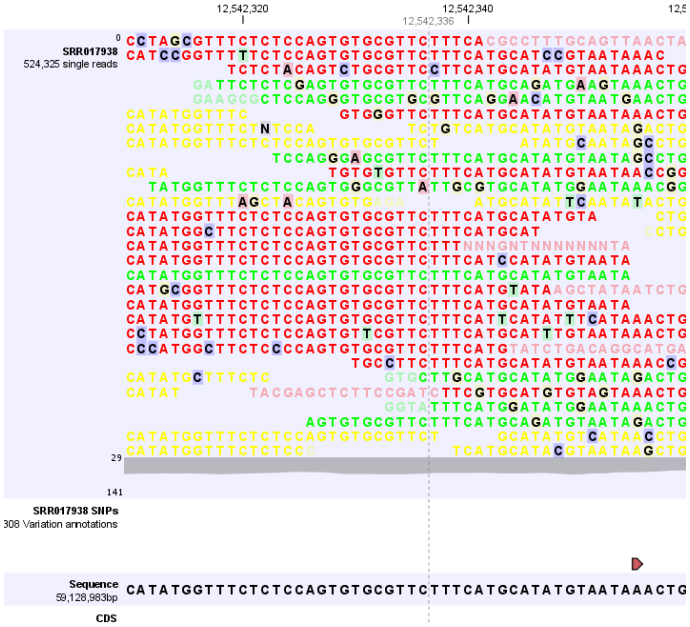


Figure 22.16: *Adjusting the height of the track.*

list can be used to jump to a different chromosome. It is also possible to jump to a new position. This can be done by typing in the start and end positions in the text fields. Thousands separators are supported. The selected region will automatically appear in the viewing area.

Insertions. Only relevant for variant tracks.

Find. Not relevant for reads tracks.

Track layout. The options for the Track layout varies depending on which track type is shown. The options for a read track are:

- **Data aggregation.** Allows you to specify whether the information in the track should be shown in detail or whether you wish to aggregate data. By aggregating data you decrease the detail level but increase the speed of the data display process, which is of particular interest when working with big data sets. The threshold (in bp) for when data should be aggregated can be specified with the drop-down box. The threshold describes the unit (or "bucket") size in base pairs, above which the data will start being aggregated. The bucket size depends on the track length and the zoom level. Hence, a data aggregation threshold with a low value will only show details when zoomed in, whereas a high value means that you can see details even when zoomed out. Please note that when using the high values, it will take longer time to display the data on the screen. Figure 22.17 shows the options for a read track and an annotation track. The data aggregation settings can be adjusted for each displayed track type.
- **Graph color.** Makes it possible to change the graph color.
- **Fix maximum of coverage graph.** Specifies the maximum coverage to be shown on the y-axis and makes the coverage on individual read tracks directly comparable with each other. Applies across all of the read mapping tracks.
- **Hide insertions below (%).** Hides insertions where the percentage of reads containing insertions is below this value. To hide all insertions, set this value to 101.
- **Highlight variants.** Variants are highlighted
- **Float variant reads to top.** When checked, reads with variations will appear at the top of the view.
- **Disconnect paired reads.** Disconnects paired end reads.
- **Show quality scores.** Shows the quality score. Ticking this option makes it possible to adjust the colors of the residues based on their quality scores. A quality score of 20 is used as default and will show all residues with a quality score of 20 or below in a blue color. Residues with quality scores above 20 will have colors that correspond to the selected color code. In this case residues with high quality scores will be shown in reddish colors. Clicking once on the color bar makes it possible to adjust the colors. Double clicking on the slider makes it possible to adjust the quality score limits. In cases where no quality scores are available, blue (the color normally used for residues with a low quality score) is used as default color for such residues.
- **Matching residues as dots.** Replaces matching residues with dots, only variants are shown in letters.
- **Show read type specific coverage.** When enabled, the coverage graph that summarizes those reads that could not be explicitly shown is now replaced by one coverage graph for each read type found in the Reads track. The read types graphs are made transparent so that the overlap can be visible when graphs are rendered on top of each other. This option can for instance be used for easy and visual comparison of the strand specific coverage.
- **Only show coverage graph.** When enabled, only the coverage graph is shown and no reads are shown.
- **Highlight reverse paired reads.** When enabled, read pairs with reverse orientation are highlighted with a light blue color.

When working with other track types such as gene tracks, other options are available:

Labels. Controls where in relation to the annotation features the labels will be shown, i.e. **Flag** places a label at the beginning of the gene, and above the feature graphics as shown in figure 22.18.

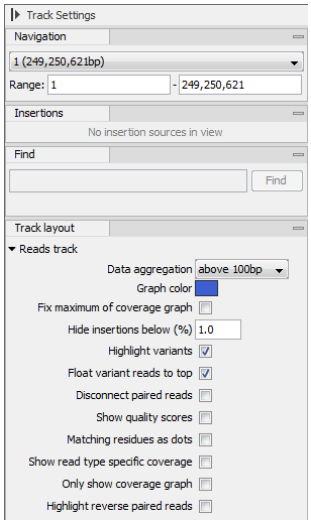


Figure 22.17: *The Side Panel for reads tracks.*

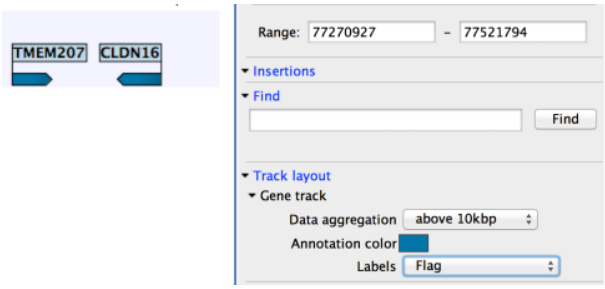


Figure 22.18: *The Side Panel for annotation tracks.*

22.3 Summary mapping report

If you choose to create a report as part of the read mapping (see section 22.2.2), this report will summarize the results of the mapping process. An example of a report is shown in figure 22.19.

The information included in the report is:

- **Summary statistics** A summary of the mapping statistics, e.g. count, percentage of reads, average length, number of bases and percentages of bases for the following type of reads:
  - References
  - Mapped reads
  - Not mapped reads
  - Reads in pairs
  - Broken paired reads

1 Mapping summary report

1.1 Summary statistics

	Count	Percentage of reads	Average length	Number of bases	Percentage of bases
References	1	-	20,158.00	20,158	-
Mapped reads	1,000	99.80%	76.00	76,000	99.89%
Not mapped reads	2	0.20%	40.00	80	0.11%
Reads in pairs	996	99.40%	250.00	75,696	99.50%
Broken paired reads	4	0.40%	76.00	304	0.40%
Total reads	1,002	100.00%	75.93	76,080	100.00%

1.2 Distribution of read length

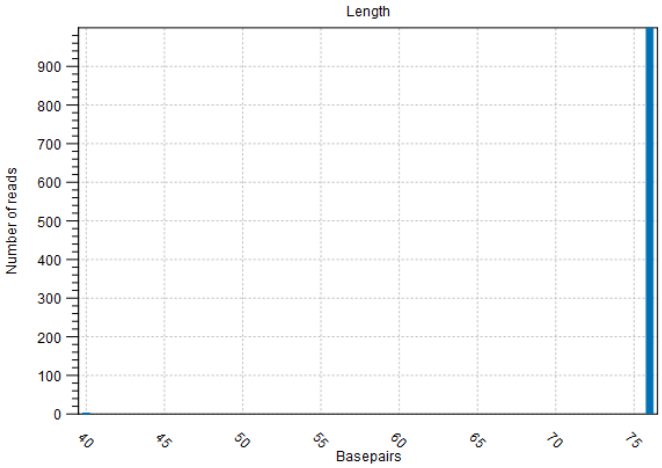


Figure 22.19: *The summary mapping report.*

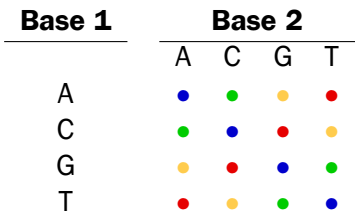
  - Total reads

- **Distribution of read length** For each sequence length, you can see the number of reads and the distribution in percent. This is mainly useful if you don't have too much variance in the lengths as you have in Sanger sequencing data for example.
- **Distribution of mapped reads length** Equivalent to the above, except that this includes only the reads that have been matched to a contig.
- **Distribution of unmapped reads lengths** Show the distribution of lengths of the unmapped sequences.
- **Paired reads distance distribution** Show the distribution of paired reads distances.

You can copy the information from the report by selecting in the report and click **Copy** (). You can also export the report in Excel format.

22.4 Mapping SOLid reads in color space

The SOLiD sequencing technology from Applied Biosystems is different from other sequencing technologies since it does not sequence one base at a time. Instead, two bases are sequenced at a time in an overlapping pattern. There are 16 different dinucleotides, but in the SOLiD technology, the dinucleotides are grouped in four carefully chosen sets, each containing four dinucleotides. The colors are as follows:

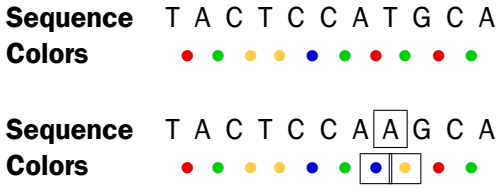


Notice how a base and a color uniquely defines the following base. This approach can be used to deduce a whole sequence from the initial nucleotide and a series of colors. Here is a sequence and the corresponding colors.

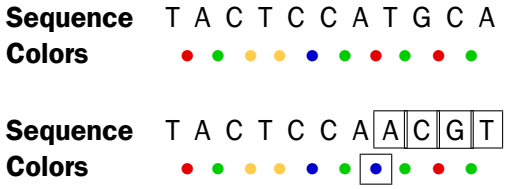
The colors do not uniquely define the sequence. Here is another sequence with the same list of colors:

But if the first nucleotide is known, the colors do uniquely define the remaining sequence. This is exactly the strategy used in SOLiD sequencing: The first nucleotide is known from the primer used, and the remaining nucleotides are deduced from the colors.

As with other sequencing technologies, errors do occur with the SOLiD technology. If a single nucleotide is changed, two colors are affected since a single nucleotide is contained in two overlapping dinucleotides:



Sometimes, a wrong color is determined at a given position. Due to the dependence between dinucleotides and colors, this affects the remaining sequence from the point of the error:



Thus, when the instrument makes an error while determining a color, the error mode is very different from when a single nucleotide is changed. This ability to differentiate different types of errors and differences is a very powerful aspect of SOLiD sequencing. With other technologies sequencing errors always appear as nucleotide differences.

22.4.1 Viewing color space information

Importing data from SOLiD systems (see section 36.2) will from *Biomedical Genomics Workbench* be imported as color space. This means that if you open the imported data, it will look like figure 22.20

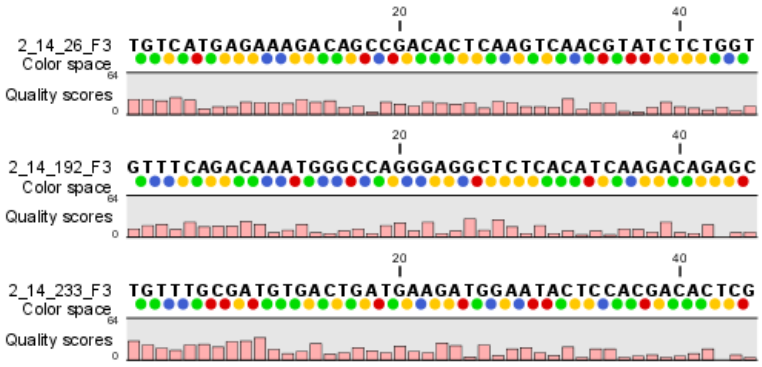


Figure 22.20: *Color space sequence list.*

In the **Side Panel** under **Nucleotide info**, you find the **Color space encoding** group which lets you define a few settings for how the colors should appear. These settings are also found in the side panel of mapping results and single sequences.

Infer encoding This is used if you want to display the colors for non-color space sequence (e.g. a reference sequence). The colors are then simply inferred from the sequence.

Show corrections This is only relevant for mapping results - it will show where the mapping process has detected color errors. An example of a color error is shown in figure 22.21.

Hide unaligned ends This option determines whether color for the unaligned ends of reads should be displayed. It also controls whether colors should be shown for gaps. The idea behind this is that these color dots will interfere with the color alignment, so it is possible to turn them off.

22.4.2 Mapping in color space

Reads from a SOLiD sequencing run may exhibit all the same differences to a reference sequence as reads from other technologies: mismatches, insertions and deletions. On top if this, SOLiD reads may exhibit color errors, where a color is read wrongly and the rest of the read is affected. If such an error is detected, it can be corrected and the rest of the read can be converted to what it would have been without the error.

Consider this SOLiD read:

Read T A C T C C A A C G T
Colors ● ● ● ● ● ● ● ● ● ●

The first nucleotide (T) is from the primer, so this is ignored in the following analysis. Now, assume that a reference sequence is this:

Reference G C A C T G C A T G C A C
Colors ● ● ● ● ● ● ● ● ● ● ● ●

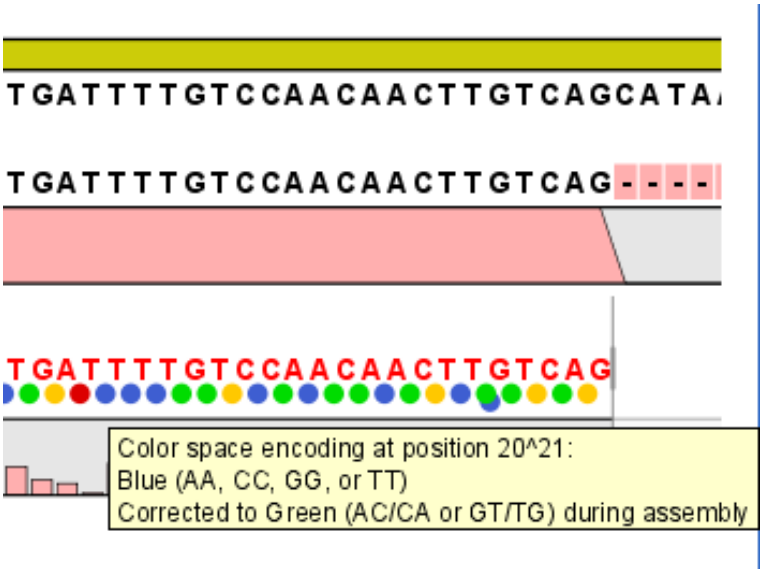


Figure 22.21: *One of the dots have both a blue and a green color. This is because this color has been corrected during mapping. Putting the mouse on the dot displays the small explanatory message.*

Here, the colors are just inferred since they are not the result of a sequencing experiment.

Looking at the colors, a possible alignment presents itself:

```
Reference   G C A C T G C A T G C A C
Colors      ● ●|●|●|●:●|●|●:●:●:●:●
                | | | : | | : : : :
Read          A C T C C A A C G T
Colors          ● ● ● ● ● ● ● ● ●
```

In the beginning of the read, the nucleotides match (ACT), then there is a mismatch (G in reference and C in read), then two more matches (CA), and finally the rest of the read does not match. But, the colors match at the end of the read. So a possible interpretation of the alignment is that there is a nucleotide change in position four of the read and a color space error between positions six and seven in the read. Such an interpretation can be represented as:

```
Reference   G C A C T G C A T G C A C
                | | | : | | | | | |
Read          A C T C C A*T G C A
```

Here, the * represents a color error. The remaining part of the displayed read sequence has been adjusted according to the inferred error. So this alignment scores nine times the match score minus the mismatch cost and a color error cost. This color error cost is a new parameter that is introduced when performing read mapping in color space.

Note that a color error may be inferred before the first nucleotide of a read. This is the very first color after the known primer nucleotide that is wrong, changing the whole read.

Here is an example from a set of real SOLiD data that was reference assembled by taking color space into account using ungapped global alignments.

```
444_1840_767_F3 has 1 match with a score of 35:

    1046535 GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA 1046569    reference
            ||||||||||||||||||||||||||||||||||
            GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA            reverse read

444_1840_803_F3 has 0 matches

444_1840_980_F3 has 1 match with a score of 29:

    2620828 GCACGAAAACGCCGCGTGGCTGGATGGT*CAAC*GTC 2620862    reference
            |||||||||||||||||||||||||||*||||*|||
            GCACGAAAACGCCGCGTGGCTGGATGGT*CAAC*GTC            read

444_1840_1046_F3 has 1 match with a score of 32:

    3673206 TT*GGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC 3673240    reference
            ||*|||||||||||||||||||||||||||||||
            TT*GGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC            reverse read

444_1841_22_F3 has 0 matches

444_1841_213_F3 has 1 match with a score of 29:

    1593797 CTTTG*AGCGCATTGGTCAGCGTGTAATCTCCTGCA 1593831    reference
            |||||*|||||||| ||||||||||||||||||||
            CTTTG*AGCGCATTAGTCAGCGTGTAATCTCCTGCA            reverse read
```

The first alignment is a perfect match and scores 35 since the reads are all of length 35. The next alignment has two inferred color errors that each count is -3 (marked by * between residues), so the score is *35 - 2 x 3 = 29*. Notice that the read is reported as the inferred sequence taking the color errors into account. The last alignment has one color error and one mismatch giving a score of *34 - 3 - 2 = 29*, since the mismatch cost is 2.

Running the same reference assembly without allowing for color errors, the result is:

```
444_1840_767_F3 has 1 match with a score of 35:

    1046535 GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA 1046569    reference
            ||||||||||||||||||||||||||||||||||
            GATACTCAATGCCGCCAAAGATGGAAGCCGGGCCA            reverse read

444_1840_803_F3 has 0 matches

444_1840_980_F3 has 0 matches

444_1840_1046_F3 has 1 match with a score of 29:

    3673206 TTGGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC 3673240    reference
              |||||||||||||||||||||||||||||||||
            AAGGTCAGGGTCTGGGCTTAGGCGGTGAATGGGGC            reverse read

444_1841_22_F3 has 0 matches
```

```
444_1841_213_F3 has 0 matches
```

The first alignment is still a perfect match, whereas two of the other alignment now do not match since they have more than two errors. The last alignment now only scores 29 instead of 32, because two mismatches replaced the one color error above. This shows the power of including the possibility of color errors when aligning: many more matches are found.

The reference assembly program in *Biomedical Genomics Workbench* does not directly support alignment in color space only, but if such an alignment was carried out, sequence 444_1841_213_F3 would have three errors, since a nucleotide mismatch leads to two color space differences. The alignment would look like this:

```
444_1841_213_F3 has 1 match with a score of 26:

    1593797 CTTTG*AGCGCATT*G*GTCAGCGTGTAATCTCCTGCA 1593831    reference
            |||||*|||||||*|*|||||||||||||||||||||
            CTTTG*AGCGCATT*G*GTCAGCGTGTAATCTCCTGCA           reverse read
```

So, the optimal solution is to both allow nucleotide mismatches and color errors in the same program when dealing with color space data. This is the approach taken by the assembly program in *Biomedical Genomics Workbench*.

Note! If you set the color error cost as low as 1 while keeping the mismatch cost at 2 or above, a mismatch will instead be represented as two adjacent color errors.

22.5 Local realignment

The goal of the local realignment tool is to improve on the alignments of the reads in an existing read mapping. The local realignment algorithm works by exploiting the information available in the alignments of *other* reads when it is attempting to re-align any given read. Most mappers do not use cross-read information as it would be computationally prohibitive to do within the mapping algorithm. However, once the reads have been mapped, local realignment procedures can exploit this information.

Realignment will typically occur in areas around insertions and deletions in the sample reads relative to the reference. In such regions we wish to see our reads mapped with one end of the read on one side of the indel and the rest mapped on the other side. However, the mapper that originally mapped the reads to the reference does not have information about the existence of an indel to use when mapping a given read. Thus, reads that are mapped to such regions, but that only have a short part of the read representing the region on one side of the indel, will typically not be mapped properly across the indel, but instead be mapped with this end unaligned, or into the indel region with many mismatches. The Local Realignment tool can use information from the other reads mapping to a region containing an indel, including reads that are located more centered across the indel and thus have been mapped with ends on either side of the indel. As a result an alternative mapping, as good as or better than the original, can be generated.

Local realignment will typically have an effect on any read mapping, whether the reads were mapped using a local or global alignment algorithm (i.e. with the Global alignment option of the mapping tool unchecked (the default) or checked, respectively). An example of the effect of using the Local Realignment tool on a read mapping made using the local alignment algorithm is shown

in figure 22.22. An example in the case of a mapping made using the global alignment algorithm is shown in figure 22.23.

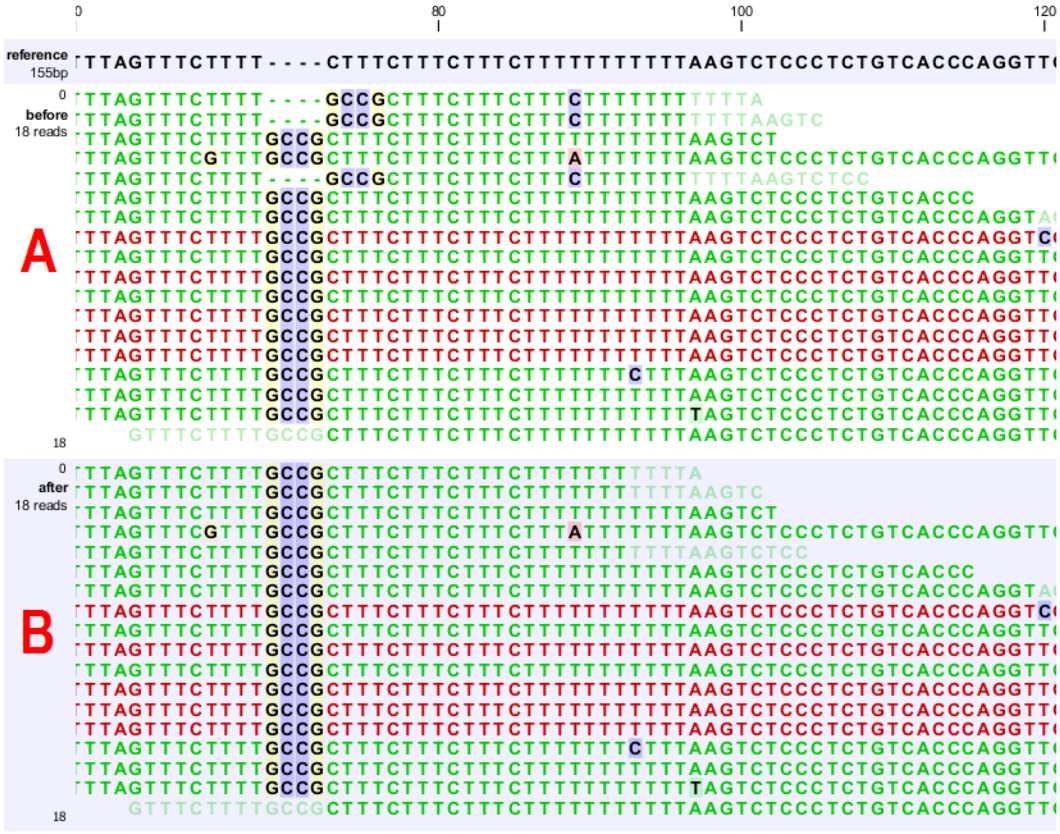


Figure 22.22: *Local realignment of a read mapping produced with the 'local' option. [A] The alignments of the first, second, and fifth read in this read mapping do not support the four-nucleotide insertion supported by the remaining reads. A variant caller might be tempted to call a heterozygous insertion of four nucleotides in one allele and heterozygous replacement of four nucleotides in a second allele. [B] After applying local realignment, the first, second, and fifth read consistently support the four-nucleotide insertion.*

22.5.1 Method

The local realignment algorithm uses a variant of the approach described by Homer et al. [Homer N, 2010]. In the first step, alignment information of all input reads are collected in an efficient graph-based data structure, which is essentially similar to a de-Brujn graph. This realignment graph represents how reads are aligned to the reference sequence and how reads overlap each other. In the second step, metadata are derived from the graph structure that indicate at which alignment positions realignment could potentially improve the read mapping, and also provides hypotheses as to how reads should be realigned to yield the most concise multiple alignment. In the third step the realignment graph and its metadata are used to actually perform the local realignment of each individual read. Figure 22.24 depicts a partial realignment graph for the read mapping shown in figure 22.22.

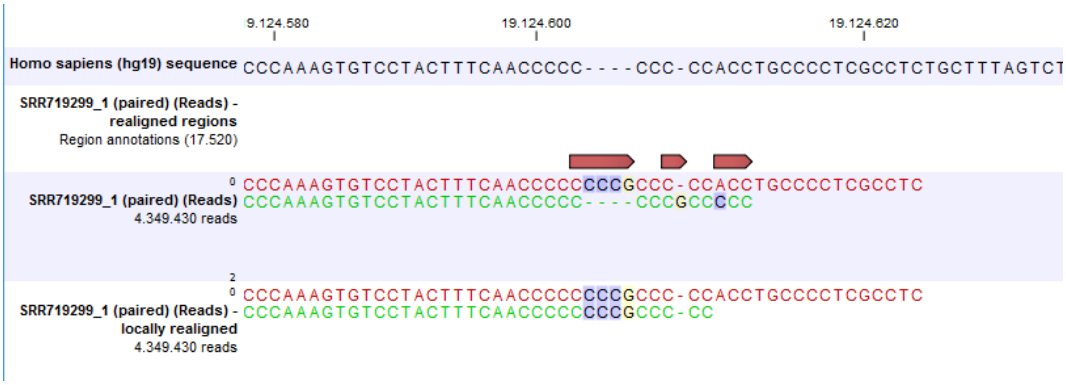


Figure 22.23: *Local realignment of a read mapping produced with the 'global' option. Before realignment the green read was mapped with two mismatches. After realignment it is mapped with the inserted 'CCCG' sequence (seen in the alignment of the red read) and no mismatches.*

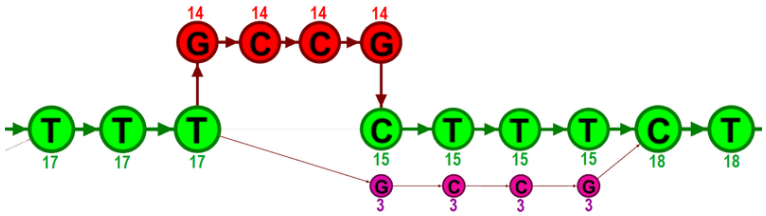


Figure 22.24: *The green nodes represent nucleotides of the reference sequence. The four red nodes represent the four-nucleotide insertion observed in fourteen mapped reads. The four violet nodes represent the four mismatches to the reference sequence observed in three mapped reads. During realignment of the original reads, two possible paths through the graph are discovered. One path leads through the four red nodes, the other through the four violet nodes. Since red nodes have been observed in fourteen of the original reads, whereas the violet nodes have only been seen in three original reads, the path through the four red nodes is preferred over the path through the violet nodes.*

22.5.2 Realignment of unaligned ends

A typical error in read alignments is the occurrence of unaligned ends (also known as soft-clipped read ends). These unaligned ends are introduced by the read mapper as a consequence of an unresolved indel towards the end of a read. Those unaligned ends can be realigned in many cases, after the read itself has been locally realigned according to the indel that prevented the read mapper from aligning the read ends correctly. Figure 22.25 depicts such an example.

22.5.3 Guided realignment

One limitation of the local realignment algorithm employed is that at least one read must be aligned correctly according to the true indel present in the data. If none of the reads is aligned correctly, local realignment cannot improve the alignment, since it lacks information about how to do so. To overcome this limitation, local realignment can be guided in two ways:

1. **Guidance variants:** By supplying the Local realignment tool with a track of guidance variants. There are two modes for using the guidance variant track: either the 'un-forced' guidance mode (if the 'Force realignment to guidance-variants' is left un-ticked) or the 'forced' guidance mode (if the 'Force realignment to guidance-variants' is ticked). In the

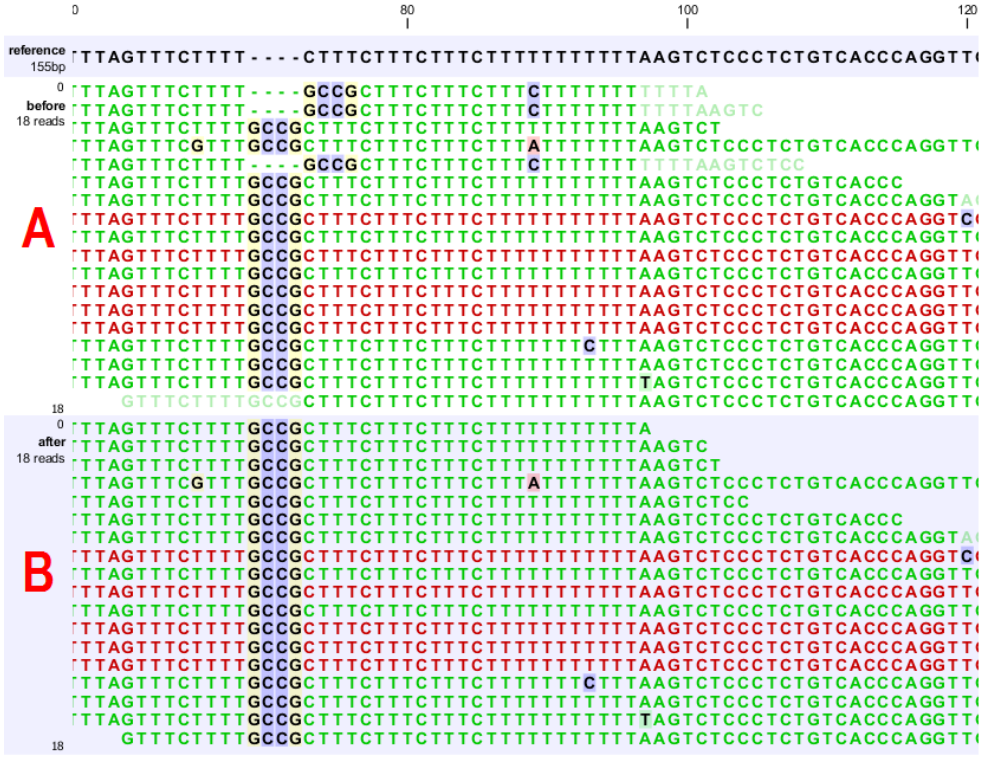


Figure 22.25: *[A] The alignments of the first, second, and fifth read in this read mapping do not support the four-nucleotide insertion supported by the remaining reads. Additionally, the first, second, fifth and the last reads have unaligned ends. [B] After applying local realignment the first, second and fifth read consistently support the four-nucleotide insertion. Additionally, all previously unaligned ends have been realigned, because they perfectly match the reference sequence now (see also figure 22.22).*

'unforced' mode, 'pseudo-reads' are given to the local realignment algorithm representing the guidance variants, allowing the local realignment algorithm to explore the paths in the graph corresponding to these alignments. In the 'forced' mode, 'pseudo-references' are given to the local realignment algorithm representing the guidance variants, allowing the reads to be aligned to allele sequences of these in addition to the original reference sequence - with matches being awarded and encouraged equally much. The 'unforced' mode can be used with any guidance variant track as input. The 'force' mode should *only* be used with guidance variants for which there is strong prior evidence that they exist in the data (e.g., the 'InDel' track from the Structural Variants' tool (see Section 22.9) produced on the read mapping that is being aligned). Unless you do have strong evidence for the presence of these guidance variants, we do not recommend using the 'forced' mode as it can lead to the introduction of false positives in your alignment and all subsequent analyses.

2. **Concurrent local realignment of multiple samples:** Multiple input read mappings increase the chance to encounter at least one read mapped correctly. This guiding mechanism has been particularly designed for scenarios, where samples are known to be related, such as in family trials.

Figure 22.26 and figure 22.27 show examples that can be improved by guiding the local realignment algorithm.

Figure 22.26: *[A] Three reads are misaligned in the presence of a four nucleotide insertion relative to the reference. [B] When applying local realignment without guidance the alignment is not improved. [C] Here local realignment is performed in the presence of the guiding variant track seen in (E). This enables the algorithm to consider alternative alignments, which are accepted whenever they have significant improvements over the original (as in read three that has a comparatively long unaligned-end). [D] If the alignment is performed with the option "Force realignment to guidance-variants" enabled, the realignment will be forced to realign according to the guiding variants track shown in (E), and this will result in realignment of all three reads. [E] The guiding variants track contains, amongst others, the four nucleotide insertion.*

22.5.4 Multi-pass local realignment

As described in section 22.5.1 the algorithm initially builds the realignment graph using the input read mapping. After the graph has been built the algorithm realigns individual reads based on information inferred from the realignment graph structure and its associated metadata. In some cases repetitive realignment iterations yield even more improvements, because with each realignment iteration the structure of the realignment graph changes slightly, potentially permitting further improvements. Local realignment therefore supports to perform multiple iterations implicitly. This is not only considered a convenience feature, but also saves a great deal of runtime by avoiding repeated transfers of large input data sets. For most samples local realignment will quickly saturate in the number of improvements. Generally, two realignment passes are strongly recommended. More than three passes rarely yield further improvements.

22.5.5 Known limitations

The major limitation of the local realignment algorithm is the necessity of at least one read being mapped correctly according to an indel present in the data. Insufficient alignment data results in suboptimal realignments or no realignments at all. As a work-around, local realignment can be guided by supplying a track of variants that enable the algorithm to determine improvements. Further guidance can be achieved by increasing the amount of alignment information and thereby increasing the chance to observe at least one read mapped correctly.

Figure 22.27: *[B] Three reads are misaligned in the presence of a four nucleotide insertion into the reference. Applying local realignment without guiding information would not yield any improvements (not shown). [C] Performing local realignment on both samples (A) and (B) enables the algorithm to improve the alignments of sample (B).*

Reads are ignored, but retained in outputs, if:

- Lengths are longer than 50,000 base pairs.
- The alignment is longer than 50,000 base pairs.
- Crossing the boundaries of circular chromosomes.

Guiding variants are ignored, if:

- They are of type "Replacement".
- They are longer than 200 bp (set as default value, but can be changed using the Maximum Guidance Variant Length parameter).
- If they are inter-chromosomal structural variations.
- If they contain ambiguous nucleotides.

22.5.6 Computational requirements

The realignment graph is produced using a sliding-window approach with a window size of 250,000 bp. If local realignment is run with multiple passes, then each pass has its own realignment graph. While memory consumption is typically below two gigabytes for single-pass, processor loads are substantial. Realigning a human sample of approximately 50x coverage will take around 24 hours on a typical desktop machine with four physical cores. Building the realignment graph and realignment of reads are parallelized actions, such that the algorithm scales very well with the number of physical cores. Server machines exploiting 12 or more physical cores typically run three times faster than the desktop with only four cores.

22.5.7 How to run the Local Realignment tool

The tool is found in the Toolbox:

Toolbox | **NGS Core Tools (**) | **Local Realignment (**)

Select one or multiple read mappings as input. If one read mapping is selected, local realignment will attempt to realign all contained reads, if appropriate. If multiple read mappings are selected, their reference genome must exactly match. Local realignment will realign all reads from all input read mappings as if they came from the same input. However, local realignment will create one output read mapping for each input read mapping, thereby preserving the affiliation of each read to its sample. Clicking Next allows you to set parameters as displayed in figure 22.28.

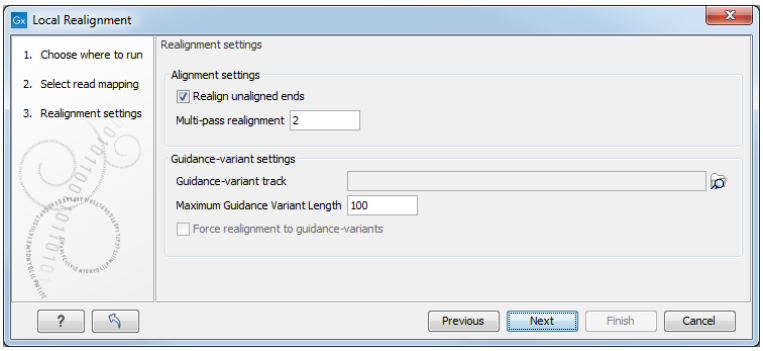


Figure 22.28: *Set the realignment options.*

Alignment settings

- **Realign unaligned ends** This option, if enabled, will trigger the realignment algorithm to attempt to realign unaligned ends as described in section "Realignment of unaligned ends (soft clipped reads)". This option should be enabled by default unless unaligned ends arise from known artifacts (such as adapter remainders in amplicon sequencing setups) and are thus not expected to be realignable anyway. Ignoring unaligned ends will yield a significant run time improvement in those cases. Realigning unaligned ends under normal conditions (where unaligned ends are expected to be realignable), however, does not contribute a lot of processing time.
- **Multi-pass realignment** This option is used to specify, how many realignment passes shall be performed by the algorithm. More passes improve accuracy at the cost of longer run time (approx. 25% per pass). Two passes are recommended; more than three passes barely yield further improvements.

Guidance-variant settings

- **Guidance-variant track** A track of variants to guide realignment of reads. Guiding can be used in at least two scenarios: (1) if reads are short or expected variants are long and (2) if cross sample comparisons are performed and some samples are already well genotyped. A track of variants can be produced by either of the variant callers, The Structural Variant tool or by importing variants from external data sources, such as dbSNP, etc.
- **Maximum Guidance Variant Length** set at 200 by default but can be increased to include guidance variants longer than 200 bp.

There are two modes for using the guidance track:

- **Un-forced** If the 'Force realignment to guidance-variants' is un-ticked the guidance variants are used as 'weak' prior evidence: each guidance variant will be represented by a pseudo-read, allowing the local realignment to explore the alignments that the guidance variants suggest. Any variant track may be used to guide the realignment when the un-forced mode is chosen.
- **Force realignment to guidance-variants** If the 'Force realignment to guidance-variants' is ticked the guidance variants are used as 'strong' prior evidence: a 'pseudo' reference will be generated for each guidance variant, and the alignment of nucleotides to their sequences will be awarded and encouraged *as much as* the alignment to the original reference sequence. Thus, the 'Force realignment to guidance-variants' options should *only* be used when there is prior information that the variants in the guidance variant track are infact present in the sample. This would e.g. be the case for an 'InDel' track produced by the Structural Variant tool (see Section 22.9), in an analysis of the same sample as the realignment is carried out on. Using 'forced' realignment to a general variant data base track is generally *strongly* discouraged.

The next dialog allows specification of the result handling. Under "Output options" it is possible to specify whether the results should be presented as a reads track or a stand-alone read mapping (figure 22.29).

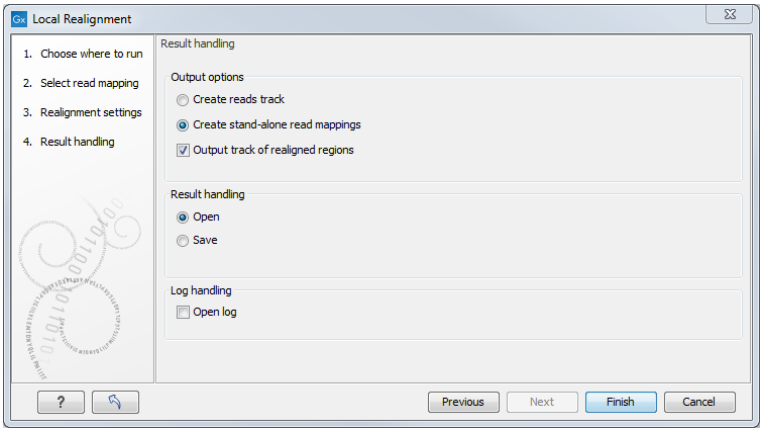


Figure 22.29: *An output track of realigned regions can be created.*

If enabled, the option **Output track of realigned regions** will cause the algorithm to output a track of regions that help pinpoint regions that have been improved by local realignment. This track has purely informative intention and cannot be used for anything else.

22.6 Merge mapping results

If you have performed two mappings with the same reference sequences, you can merge the results using the **Merge Read Mappings** (). This can be useful in situations where you have already performed a mapping with one data set, and you receive a second data set that you want to have mapped together with the first one. In this case, you can run a new mapping of the second data set and merge the results:

Toolbox | **Resequencing Analysis ()** | **Merge Read Mappings ()**

This opens a dialog where you can select two or more mapping results, either in the form of tracks or read mappings. If the mappings are based on the same reference sequences (based on the name and length of the reference sequence), the reads will be merged into one mapping. If different reference sequences are used, they will simply be be incorporated into the same result file (either a track or a mapping table).

The output from the merge can either be a track or standard mappings (equivalent to the read mapper's output, see section 22.1). For all the mappings that could be merged, a new mapping will be created. If you have used a mapping table as input, the result will be a mapping table. Note that the consensus sequence is updated to reflect the merge. The consensus voting scheme for the first mapping is used to determine the consensus sequence. This also means that for large mappings, the data processing can be quite demanding for your computer.

22.7 Remove duplicate mapped reads

The purpose of this tool is to efficiently remove duplicate reads from a mapping, when duplicate reads have arisen due to the use of PCR amplification (or other enrichment) during sample preparation. This does not mean, however, that this tool should be used on all data that had an amplification step. In fact, use of this tool in the case of RNA-Seq data, amplicon data, or any sample where the start of a large number of reads are purposely at the same reference location, it is not recommended to make use of this tool. The tool may be used on mappings of single end reads, paired end reads or both.

A read duplication can be easily distinguished when mapping reads to a reference sequence as shown in the example in figure 22.30.

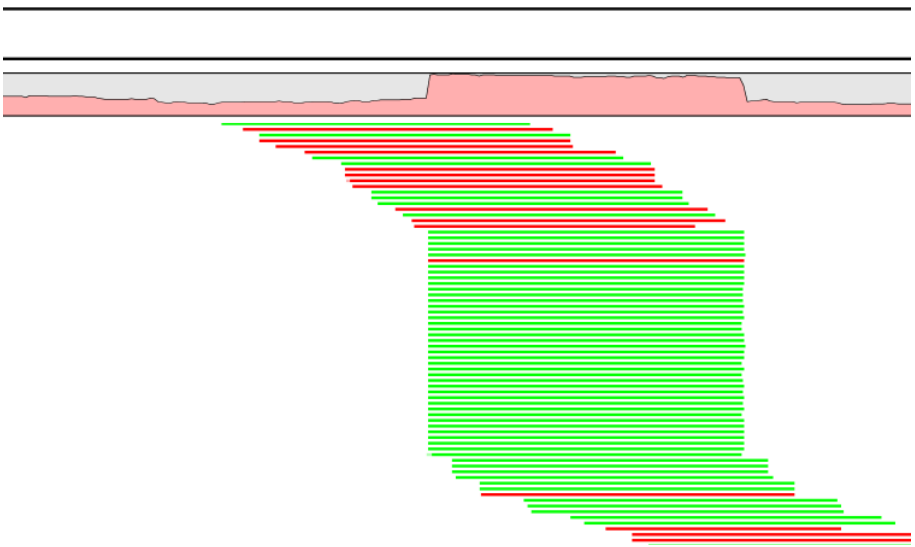


Figure 22.30: *Mapped reads with a set of duplicate reads, the colors denote the strand (green is forward and red is reverse).*

When sequencing library preparation involves a PCR amplification step, it is common to observe multiple reads where identical nucleotide sequences are disproportionably represented in the final results. Thus, to facilitate processing of mappings based on this kind of data, it may be necessary to perform a duplicate read removal step, which flags identical reads and subsequently

removes them from the data set. However, this step is complicated by the low, but consistent, presence of sequencing errors that may cause otherwise identical sequences to differ slightly. Thus, it is important that the duplicate read removal includes some tolerance for nearly identical sequences, which could still be reads from the same PCR artifact.

In samples that have been mapped to a reference genome, duplicate reads from PCR amplification typically result in areas of disproportionally high coverage and are often the cause of significant skew in allelic ratios, particularly when replication errors are made by the enzymes (e.g. polymerases) used during amplification. Sequencing errors incorporated post-amplification can affect both sequence- and coverage-based analysis methods, such as variant calling, where introduced errors can create false positive SNPs, and ChIP-Seq, where artificially inflated coverage can skew the significance of certain locations. By utilizing the mapping information, it is possible to perform the duplicate removal process rapidly and efficiently.

Note! We only recommend using the duplicate read removal if there are amplification steps involved in the library preparation. It is not reccomended for RNA-Seq data, amplicon data, or any sample where the start of a large number of reads are purposely at the same reference location.

The method used by the duplicate read removal is to identify reads that share common coordinates (e.g. the same start and end coordinate), sequencing direction (or mapped strand) and the same sequence, these being the unifying characteristics behind sequencing reads that originate from the same amplified fragments of nuclear material. However, due to the frequent occurrence of sequencing errors, the tool utilizes simple heuristics to prune sequences with small variations from the consensus, as would be expected from errors observed in data from next-generation sequencing platforms. Base mismatch errors that were incorporated during amplification or prior to amplification will be indistinguishable from SNPs and may not be filtered out by this tool.

22.7.1 Algorithm details and parameters

The algorithm operates with the following assumption: Mapped reads from duplicated DNA fragments will share a mapping orientation (e.g. will map to the same strand), and depending on their orientation, will share either a start coordinate (forward reads), an end coordinate (reverse reads) or both (paired end reads).

Based on this assumption, a group of reads that share identical start and end coordinates (or start coordinate and length for single end reads) and also share identical sequences can be considered as potential duplications of the same DNA fragment. These reads are then investigated to find reads to be removed and reads to be kept. In order to explain how this works, we will use a small example shown in figure 22.31.

```
ACGGACTGCTT    60
ACTGACTGCTT     5
ACTGACTGATT   100
```

Figure 22.31: *An alignment of three different sequence. The numbers are read counts, e.g. the read at the top occurs 60 times.*

The example shows 165 reads that share the same start position and orientation and are considered for duplicate read removal. 60 reads share the sequence shown at the top, 5 reads

share the middle sequence, and 100 reads share the sequence at the bottom. The differences are at position 3 and 9 (underlined).

The tool will now create a tree structure out of these reads as illustrated in figure 22.32.

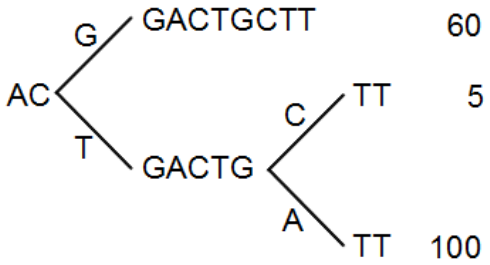


Figure 22.32: *The reads from figure 22.31 represented as a Patricia tree [Morrison, 1968].*

The first branch point in the tree is at the third position, where sequence number one has a G and the other sequences have a T. The other two sequence disagree at position nine, where one has a C and another has an A.

The next step is to iteratively merge the branches, starting from the end of the tree. The first branch point to consider is at position nine. Since only 5 reads have a C and 100 reads have an A, the C branch is collapsed. This is shown in figure 22.33.

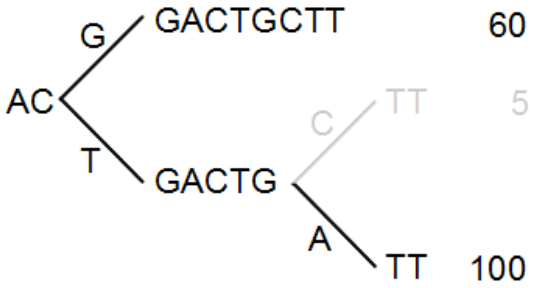


Figure 22.33: *Merging the sequences.*

As a user, you can specify the **threshold** for when the reads should be merged. The default is 20 %: when the minority branch has less than 20 % of the read count of the both branches, it is collapsed.

The next branch to consider is at the third position, where there are now 105 reads that have a T and 60 reads that have a G (see figure 22.34).

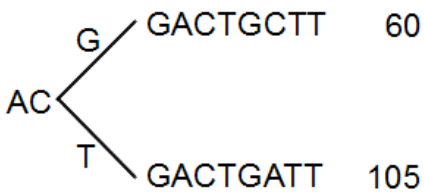


Figure 22.34: *Merging the sequences.*

With the default setting at 20%, these two branches will not be collapsed, because there are too many reads on the minority branch (60 reads versus 105 reads). Since this process is aimed at collapsing reads that are only distinguished apart by sequencing errors, you would not expect this situation to be caused by sequencing errors, but rather true biological variation (or PCR errors in the early cycles that are indistinguishable from true variation).

If we raised the threshold to 60%, the two branches above would be merged into one if it was not for the second rule governing the merging of branches: The sequences have to be identical except for the difference at the branch point. Looking at the sequences in figure 22.34, there is a difference at position 9 which means that these two branches would never be merged, regardless the threshold and the read counts.

The result of the duplicate reads removal in this example would be that the 165 reads are reduced to two in the result.

22.7.2 Running the duplicate reads removal

The tool is found in the Toolbox:

Toolbox | **Resequencing Analysis** ( ) | **Remove Duplicate Mapped Reads** ( )

This opens a dialog where you can select mapping results in read tracks ( ) format. Clicking **Next** allows you to set the threshold parameters as displayed in figure 22.35.

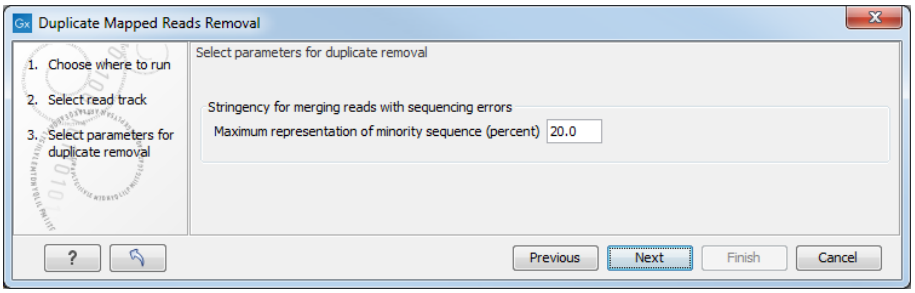


Figure 22.35: *Setting the stringency for merging similar reads.*

The parameter is explained in detail in section 22.7.1.

Clicking **Next** will reveal the output options. The main output is a list of the reads that remain after the duplicates have been removed. In addition, you can get the following output:

List of duplicate sequences These are the sequences that have been removed.

Report This is a brief summary report with the number of reads that have been removed (see an example in figure 22.36).

Note! The Remove Duplicate Mapped Reads tool may run this before or after local realignment. The order in which these two tools are run should make little if any difference.

22.8 Extract reads based on overlap

This tool can be used to extract subsets of reads based on annotations. When extracting reads with a specific annotation, the annotation will function as a tag pulling out all the reads with the

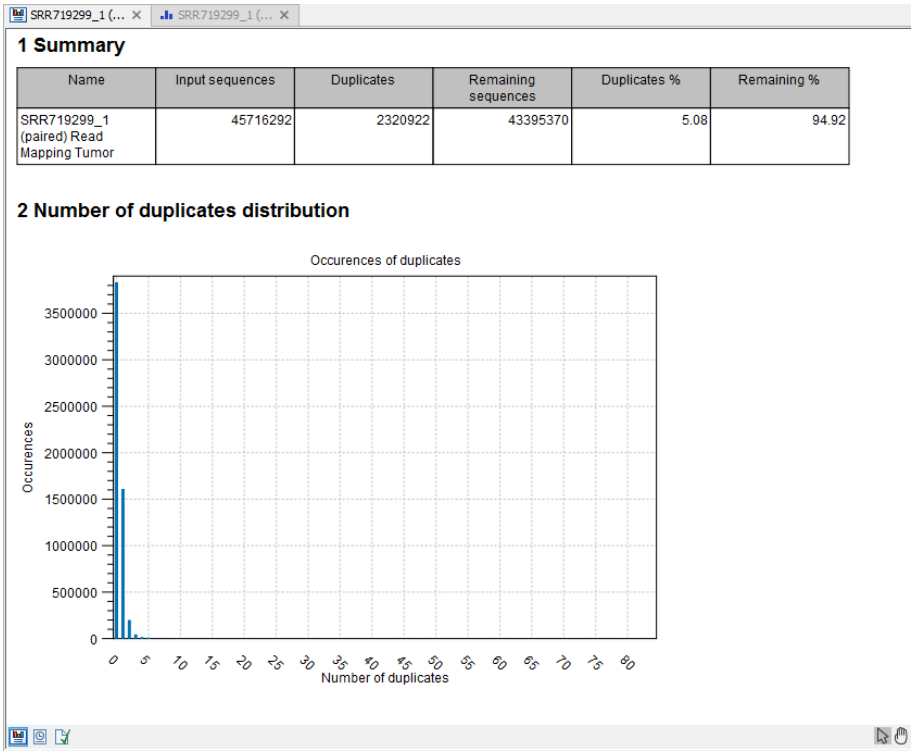


Figure 22.36: *Summary statistics on the duplicate mapped reads.*

overlapping annotation (or, when handling paired read data, all the pairs of reads). To launch the tool, go to:

Toolbox | **Track Tools** ( ) | **Annotate and Filter** | **Extract Reads Based on Overlap** ( )

Read mapping tracks can be used as input (figure 22.37).

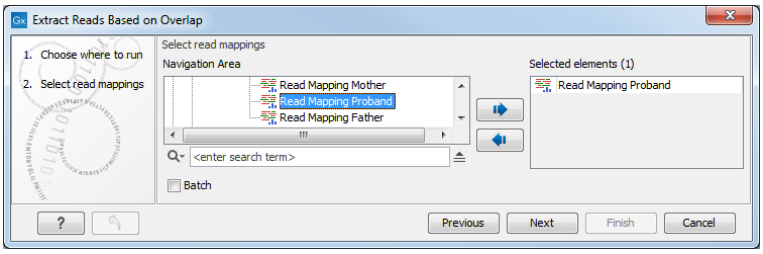


Figure 22.37: *Select a read mapping. Only one read mapping can be selected at the time.*

The next step is to select the annotated track(s) to be used for pulling out reads and specify which reads to include (figure 22.38). Note that it is also possible to select here a RNA-seq statistical comparison.

The options in this wizard are:

Overlap tracks

- Select the annotated track

reads within the intervals It is possible to select whether only reads within the intervals should be extracted, or whether reads continuing outside the annotated region should be extracted. The difference between the options can be seen in figure 22.39.

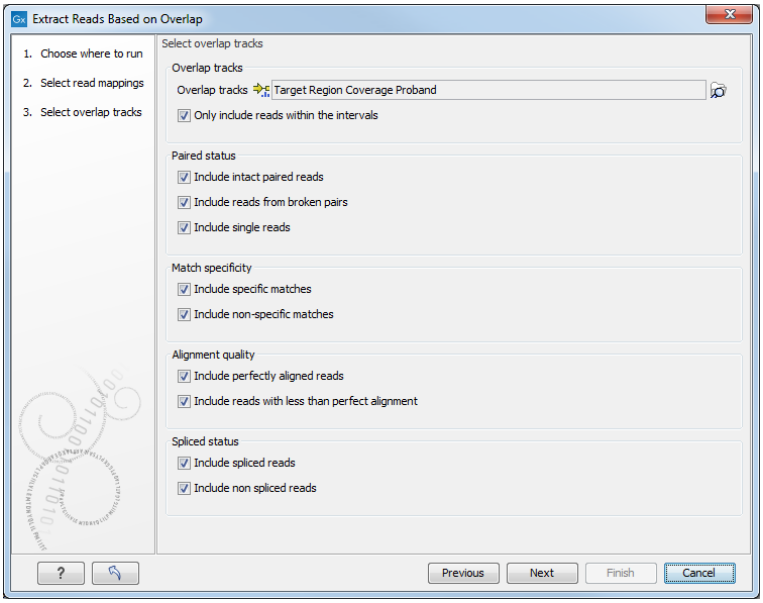


Figure 22.38: *Select the track(s) containing the annotation(s) of interest. Multiple tracks can be selected at the same time.*

Paired status

clude 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.

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.

ude non-specific matches Reads that have multiple equally good alignments to the reference. These reads are colored yellow per default.

Alignment quality

e 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.

s 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.

nclude non spliced reads Reads that are not across an intron.

Figure 22.38 shows two examples of output when the overlap track used as input was generated using the "Identify Graph Threshold Areas".

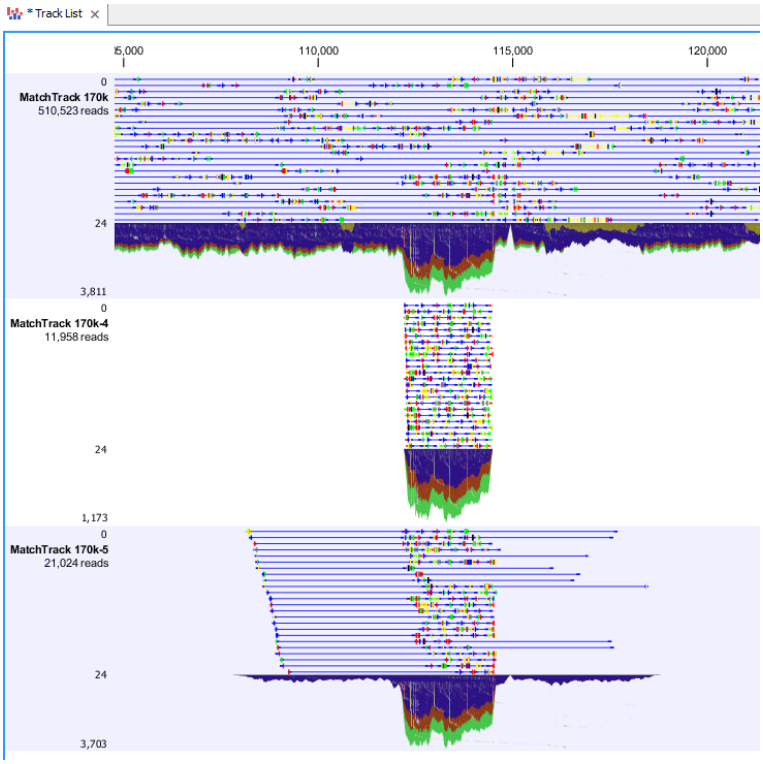


Figure 22.39: *Output from Extract reads based on overlap. Top: The read mapping used as input. Middle: Output when "Only include reads within intervals" has been ticked. Bottom: Output when "Only include reads within intervals" has been deselected.*

22.9 InDels and Structural Variants

The InDels and Structural Variants tool is designed to identify structural variants such as insertions, deletions, inversions, translocations and tandem duplications in read mappings. The tool relies *exclusively* on information derived from unaligned ends (also called 'soft clippings') of the reads in the mappings. This means that:

- The tool will detect NO structural variants if there are NO reads with unaligned ends in the read mapping.
- Read mappings made with the CLC 'Map reads to reference' tool with the *'global'* option switched on will have NO unaligned ends and the InDels and Structural Variants tool will thus find NO structural variants on these. (The 'global' option means that reads are aligned in their entirety - irrespectively of whether that introduces mismatches towards the ends of the reads. In the 'local' option such reads will be mapped with unaligned ends).
- Read mappings based on really short reads (say, below 35 bp) are not likely to produce many reads with unaligned ends of any useful length, and the tool is thus not likely to produce many structural variant predictions for these read mappings.

- Read mappings generated with the Large Gap Read Mapper are NOT optimal for the detection of structural variants with this tool. This is due to the fact that, the Large Gap Read Mapper will map some reads with (large) gaps, that would be mapped with unaligned ends with standard read mappers, and thus will leave a weaker unaligned end signal in the mappings for the Structural Variation tool to work with.

In its current version the InDels and Structural Variants tool has the following known limitations:

- It will only detect intra-chromosomal structural variants.

22.9.1 How to run the InDels and Structural Variants tool

To start the structural variant detection:

Toolbox | **Resequencing** () | **InDels and Structural Variants tool** ()

This will open up a dialog. Select the read mapping of interest as shown in figure 22.40 and click on the button labeled **Next**.

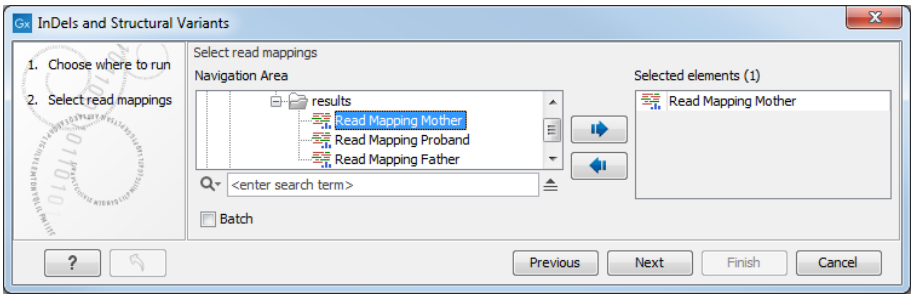


Figure 22.40: *Select the read mapping of interest.*

The next wizard step (Figure 22.41) is concerned with specifying parameters related to the algorithm used for calling structural variants. The algorithm first identifies positions in the mapping(s) with an *excess* of reads with left (or right) unaligned ends. Once these positions and the consensus sequences of the unaligned ends are determined, the algorithm maps the determined consensus sequences to the reference sequence around other positions with unaligned ends. If mappings are found that are in accordance with a 'signature' of a structural variant, a structural variant is called. For further details about the algorithm see section 22.9.3.

The **'Significance of unaligned end breakpoints'** parameters are concerned with when a position with unaligned ends should be considered by the algorithm, and when it should be ignored:

- **P-value threshold**: Only positions in which the fraction of reads with unaligned ends is sufficiently high will be considered. The 'P-value threshold' determines the cut-off value in a Binomial Distribution for this fraction. The higher the P-value threshold is set, the more unaligned breakpoints will be identified.
- **Maximum number of mismatches**: The 'Maximum number of mismatches' parameter determines which reads should be considered when inferring unaligned end breakpoints. Poorly map reads tend to have many mis-matches and unaligned ends, and it may be preferable to let the algorithm ignore reads with too many mis-matches in order to avoid false positives and reduce computational time. On the other hand, if the allowed number of mis-matches is set too low, unaligned end breakpoints in proximities of other variants

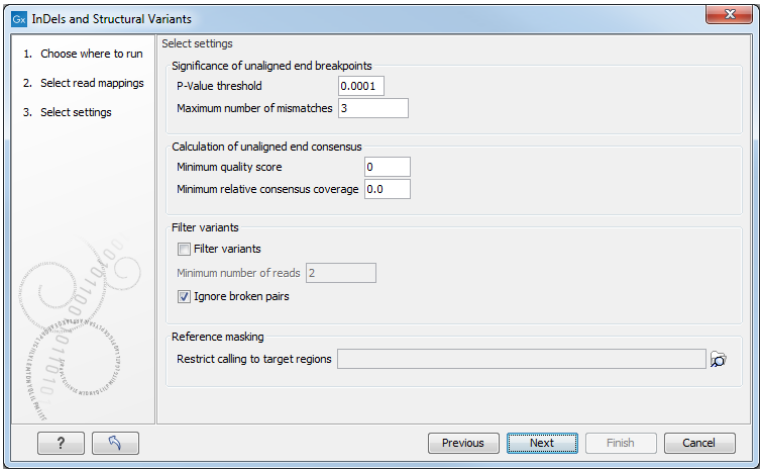


Figure 22.41: *Select the relevant settings.*

(e.g. SNVs) may be lost. Again, the higher the number of mis-matches allowed, the more unaligned breakpoints will be identified.

The **Calculation of unaligned end consensus** parameters can improve the calculation of the unaligned end consensus by removing bases according to:

- **Minimum quality score**: quality score under which bases should be ignored.
- **Minimum relative consensus coverage**: consensus coverage threshold under which bases should be ignored. The relative consensus coverage is calculated by taking the coverage at the current nucleotide position and dividing by the maximum coverage obtained along the unaligned ends upstream from this position. When the value thus calculated falls below the specified threshold, consensus generation stops. The idea behind the "Minimum relative consensus coverage" option is to stop consensus generation when dramatic drops in coverage are observed. For example, a drop from 1000 coverage to 10 coverage would give a relative consensus coverage of 10/1000 = 0.01.

The **'Filter variants'** parameters are concerned with the amount of evidence for each structural variant required for it to be called:

- **Filter variants**: When the **Filter variants** box is checked, only variants that are inferred by breakpoints that *together* are supported by at least the specified **Minimum number of reads** will be called.
- **Ignore broken pairs**: This option is checked by default, but it can be unchecked to include variants located in broken pairs.

'Reference masking' allows specification of target regions:

- **Restrict calling to target regions**: When specifying a target region track only reads that overlap with at least one of the targets will be examined when the unaligned end breakpoints are identified. Hence only breakpoints that fall within, or in close proximity of, the targets will be identified (a read may overlap a target, but have an unaligned end outside the target - these are also identified and therefore breakpoints outside, but in the proximity of the

target). The runtime will be decreased when you specify a target track as compared to when you do not.

Note! As the set of identified unaligned end breakpoints differs between runs where a target region track has been specified and where it has not, the set of predicted indels and structural variants is also likely to differ. This is because the indels and structural variants are predicted from the mapping patterns of the unaligned ends at the set of identified breakpoints. This is also the case even if you restrict the comparison to only involve the indels and structural variants detected within the target regions. You cannot expect these to be exactly the same but you can expect a large overlap.

Specify these settings and click **Next**. The "Results handling" dialog (Figure 22.42) will be opened. The Indels and Structural variants tool has the following output options:

- **Create report** When ticked, a report that summarizes information about the inferred breakpoints and variants is created.
- **Create breakpoints** When ticked, a track containing the detected breakpoints is created.
- **Create InDel variants** When ticked, a variant track containing the detected indels that fulfill the requirements for being 'variants' is created. These include:
  - the detected insertions for which the allele sequence is inferred, but not those for which it is not known, or only partly known. As the algorithm relies on mapping two unaligned ends against each other for detecting insertions *with inferred allele sequence*, the maximum length of these that can potentially be detected depends on (1) the read length and (2) the "length fraction" parameter of the read mapper. With current read lengths and default settings you are unlikely to get insertions *with inferred allele sequence* larger than a couple of hundred, and hence will not see insertions in this track larger than that.
  - medium sized deletions (those between six and 200 bp). All other deletions are put in the "Structural variants" track. The reason for not including all detected deletions in the indel track is that the main intended use of this track is to guide re-alignment. In our experience, the re-alignment algorithm performs best when only including the medium sized events. Notice that, in contrast to insertions, there is no upper limit on the length of deletions *with inferred allele sequence* that the algorithm can detect. This is because the allele sequence is trivial for deletions, whereas for insertions it must be inferred from the nucleotides in the unaligned ends.

  See section 22.20.1 for a definition of the requirements for 'variants'. Note that insertions and deletions that are not included in the InDel track, will be present in the 'Structural variants track' (described below).
- **Create structural variations** When ticked, a track containing the detected structural variants is created, including the insertions *with unknown allele sequence* and the deletions that are not included in the "InDel" track.

An example of the output from the InDels and Structural Variant tool is shown in Figure 22.43. The output is described in detail in the next section (section 22.9.2).

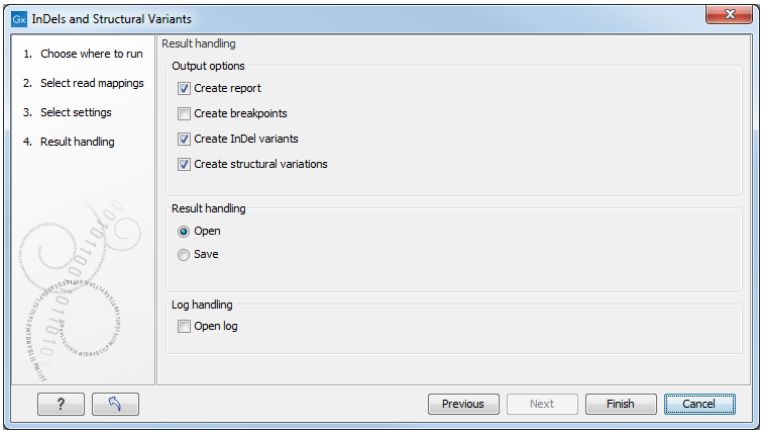


Figure 22.42: *Select output formats.*

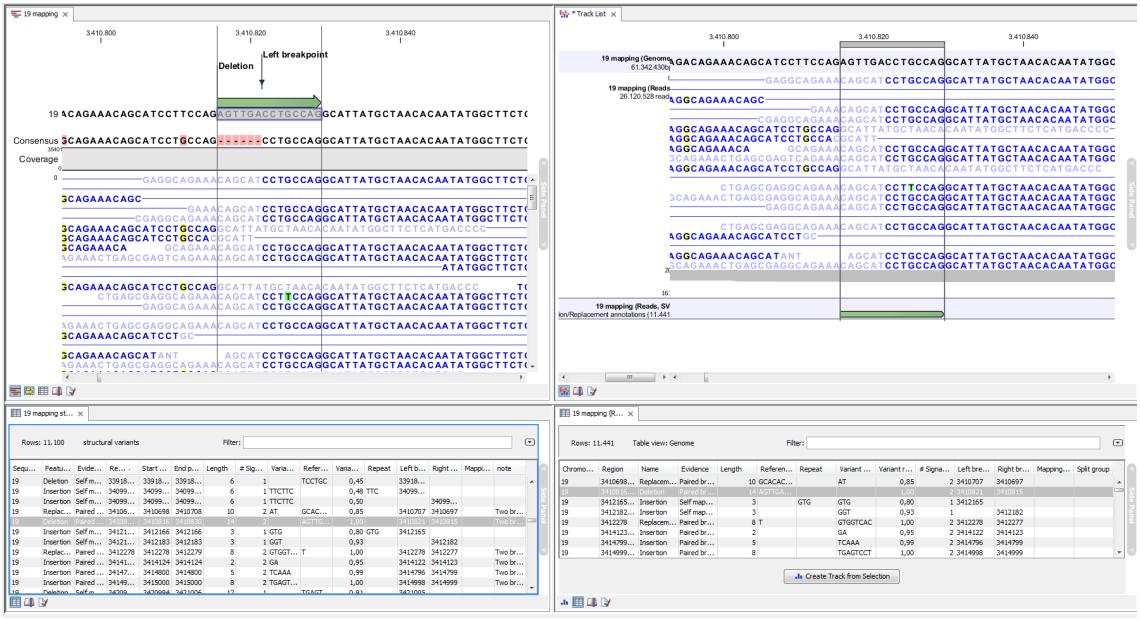


Figure 22.43: *Example of the result of an analysis on a standalone read mapping (to the left) and on a reads track (to the right).*

22.9.2 The Structural Variants and InDels output

The report

The report gives an overview of the numbers and types of structural variants found in the sample. It contains

- A table with a row for each reference sequence, and information on the number of breakpoint signatures and structural variants found.
- A table giving the total number of left and right unaligned end breakpoint signatures found, and the total number of reads supporting them.
- A distribution of the logarithm of the sequence complexity of the unaligned ends of the left and right breakpoint signatures (see section 22.9.7 for how the complexity is calculated).
- A distribution of the length of the unaligned ends of the left and right breakpoint signatures.

- A table giving the total number of the different types of structural variants found.
- Plots depicting the distribution of the lengths of structural variants identified.

The Breakpoints track (BP):

The breakpoints track contains a row for each called breakpoint with the following information:

- **Chromosome**: The chromosome on which the breakpoint is located.
- **Region**: The location on the chromosome of the breakpoint.
- **Name**: The type of the breakpoint ('left breakpoint' or 'right breakpoint').
- **p-value**: The p-value (in the Binomial distribution) of the unaligned end breakpoint.
- **Unaligned**: The consensus sequence of the unaligned ends at the breakpoint.
- **Unaligned length**: The length of the consensus sequence of the unaligned ends at the breakpoint.
- **Mapped to self**: If the unaligned end sequence at the breakpoint was found to map back to the reference in the vicinity of the breakpoint itself, a 'Deletion' or 'Insertion' based on 'self-mapping' evidence is called. This column will contain 'Deletion' or 'Insertion' if that is the case, or be empty if the unaligned end did not map back to the reference in the vicinity of the breakpoint itself.
- **Perfect mapped**: The number of 'perfect mapped' reads. This number is intended as a proxy for the number of reads that fit with the reference sequence. When calculating this number we consider all reads that extend across the breakpoint. We ignore reads that are non-specifically mapped, in a broken pair, or has more than the **maximum number of mismatches**. A read is perfectly mapped if (1) it has no insertions or deletions (mismatches are allowed) and (2) it has no unaligned end.
- **Not perfect mapped**: The number of 'not perfect mapped' reads. This number is intended as a proxy for the number of reads that fit with the predicted indel. When calculating this number we consider all reads that extend across the breakpoint or that has an unaligned end starting at the breakpoint. We ignore reads that are non-specifically mapped, in a broken pair, or has more than the **maximum number of mismatches**. A read is not perfect mapped if (1) it has an insertion or deletion or (2) it has an unaligned end.
- Fraction non-perfectly mapped': the 'Non perfect mapped' divided by the 'Non perfect mapped' + 'Perfect mapped'.
- **Sequence complexity**: The sequence complexity of the unaligned end of the breakpoint (see section 22.9.7 for how the sequence complexity is calculated).
- **Reads**: The number of reads supporting the breakpoint.

Note that typically, breakpoints will be found for which it is not possible to infer a structural variant. There may be a number of reasons for that: (1) the unaligned ends from which the breakpoint signature was derived might not be caused by an underlying structural variant, but

merely be due to read mapping issues or noise, or (2) the breakpoint(s) which the detected breakpoint should have been matched to was/were not detected, and therefore no matching breakpoint(s) were found. Breakpoints may go un-detected either because of lack of coverage in the breakpoint region or because they are located within regions with exclusively non-uniquely mapped reads (only unaligned ends of uniquely mapping reads are used).

The InDel variants track (InDel):

The Indel variants track contains a row for each of the called insertions or deletions that *fulfills the requirements for being of a 'variant' type* (see section 22.20.3 for a description of the variant types "Insertion" and "Deletion"). These are the small to medium sized insertions and deletions (up to 200 bp in length) for which the algorithm was able to identify the allele sequence (that is, the exact inserted sequence, or the exact deleted sequence). For insertions, the full allele sequence is found from the unaligned ends of mapped reads. For some insertions the length and allele sequence cannot be determined and as these do not fulfill the requirements of a 'variant', they do not qualify for representation in the 'InDel variant' track but instead appear in the Structural Variants track (see below). The information provided for each of the indels in the InDel variant track is the 'Chromosome', 'Region', 'Type', 'Reference', 'Allele', 'Reference Allele', 'Length' and 'Zygosity' columns that are provided for all variants (see section 22.20.1). In addition the following information, which is primarily intended to allow the user to assess the degree of evidence supporting each predicted indel, is provided:

- **Evidence**: The mapping evidence on which the call of the indel was based. This may be either 'Self mapped', 'Paired breakpoint', Cross mapped breakpoint' or 'Tandem duplication' depending of the mapping signature of the unaligned ends of the breakpoint(s) from which the indel was inferred.
- **Repeat**: The algorithm attempts to identify if the variant sequence contains perfect repeats. This is done by searching the region around the structural variant for perfect repeat sequences. The region searched is 3 times the length of variant around the insertion/deletion point. The maximum repeat length searched for is 10. If a repeat sequence is found, the repeated sequence is given in this column. If not, the column is empty.
- **Variant ratio**: This column contains the sum of the 'Non perfect mapped' reads for the breakpoints used to infer the indel, divided by the sum of the 'Non perfect mapped' and 'Perfect mapped' reads for the breakpoints used to infer the indel (see section the description above of the breakpoints track). This fraction is intended to give a hint towards the zygosity of the indel. The closer the value to 1, the higher the likelihood that the variant is homozygous.
- **# Reads**: The total number of reads supporting the breakpoints from which the indel was constructed.
- **Sequence complexity**: The sequence complexity of the unaligned end of the breakpoint (see section 22.9.7). Indels with higher complexity are typically more reliable than those with low complexity.

The 'Zygosity' field is set to 'Homozygous' if the 'Variant ratio' is 0.80 or above, and 'Heterozygous' otherwise.

The Structural variants track (SV):

The Structural variants track contains a row for each of the called Structural variants that is not already reported in the InDel track. It contains the following information:

- **Chromosome**: The chromosome on which the structural variant is located.
- **Region**: The location on the chromosome of the structural variant.
- **Name**: The type of the structural variant ('deletion', 'insertion', 'inversion', 'replacement', 'translocation' or 'complex').
- **Evidence**: The breakpoint mapping evidence ('that is, the 'unaligned end 'signature') on which the call of the structural variant was based. This may be either 'Self mapped', 'Paired breakpoint', 'Cross mapped breakpoints', 'Cross mapped breakpoints (invalid orientation)', 'Close breakpoints', 'Multiple breakpoints' or 'Tandem duplication', depending on which type of signature that was found.
- **Length**: the length of the allele sequence of the structural variant. Note that the length of variants for which the allele sequence could not be determined is reported as 0 (e.g insertions inferred from 'Close breakpoints').
- Reference sequence': The sequence of the reference in the region of the structural variant.
- **Variant sequence**: The allele sequence of the structural variant if it is known. If not, the column will be empty.
- **Repeat**: The same as in the InDel track.
- **Variant ratio**: The same as in the InDel track.
- **Signatures**: The number of unaligned breakpoints involved in the signature of the structural variant. In most cases these will be pairs of breakpoints, and the value is 2, however some structural variants that have signatures involving more than two breakpoint (See section 22.9.6). Typically structural variants of type 'complex' will be inferred from more than 2 breakpoint signatures.
- **Left breakpoints**: The positions of the 'Left breakpoints' involved in the signature of the structural variant.
- **Right breakpoints**: The positions of the 'Right breakpoints' involved in the signature of the structural variant.
- **Mapping scores fraction**: The mapping scores of the unaligned ends for each of the breakpoints. These are the similarity values between the unaligned end and the region of the reference to which it was mapped. The values lie between 0 and 1. The closer the value is to 1, the better the match, suggesting better reliability of the inferred variant.
- **Reads**: The total number of reads supporting the breakpoints from which the indels was constructed.
- **Sequence complexity**: The sequence complexity of the unaligned end of the breakpoint (see section 22.9.7).

- **Split group**: Some structural variants extend over a very large a region. For these visualization is challenging, and instead of reporting them in a single row we split them in multiple rows - one for each 'end' of the variant. To allow the user to see which of these 'split features' belong together, we give features that belong to the same structural variant a common 'split group' identifier. If the column is empty the structural variant is not split, but contained within a single row.

22.9.3 The InDels and Structural Variants detection algorithm

The Indels and Structural Variants detection algorithm has two steps:

1. Identify 'breakpoint signatures': First, the algorithm identifies positions in the mapping(s) with an *excess* of reads with left (or right) unaligned ends. For each of these, it creates a Left breakpoint (LB) or Right breakpoint (RB) signature.
2. Identify 'structural variant signatures': Secondly, the algorithm creates structural variant signatures from the identified breakpoint signatures. This is done by mapping the consensus unaligned ends of the identified LB and RB signatures to selected areas of the references as well as to each other. The mapping patterns of the consensus unaligned ends are examined and structural variant annotations consistent with the mapping patterns are created.

The two steps of the algorithm are described in detail in sections 22.9.4 and 22.9.5.

22.9.4 The InDels and Structural Variants detection algorithm - Step 1: Creating Left- and Right breakpoint signatures

In the first step of the InDels and Structural Variants detection algorithm points in the read mapping are identified which have a significant proportion of reads mapped with unaligned ends. There are typically numerous reads with unaligned ends in read mappings — some are due to structural variants in the sample relative to the reference, others are due to poorly mapped, or poor quality reads. An example is given in figure 22.44. In order to make reliable predictions, attempts must be made to distinguish the unaligned ends caused by noisy read(mappings) from those caused by structural variants, so that the signal from the structural variants comes through as clearly as possible — both in terms of where the 'significant' unaligned ends are and in terms of what they look like.

To identify positions with a 'significant' portion of 'consistent' unaligned end reads we first estimate 'null-distributions' of the fractions of left and right unaligned end reads at each position in the read mapping, and subsequently use these distributions to identify positions with an 'excess' of unaligned end reads. In these positions we create a Left (LB) or Right (RB) breakpoint signature. To estimate the null-distributions we:

1. Calculate the coverage, c_i, in each position, i of all uniquely mapped reads (Non-specifically mapped reads are ignored. Furthermore, for paired read data sets, only intact paired reads pairs are considered — broken paired reads are ignored).
2. Calculate the coverage in each position of 'valid' reads with a starting left unaligned end, l_i (of minimum consensus length 3bp).

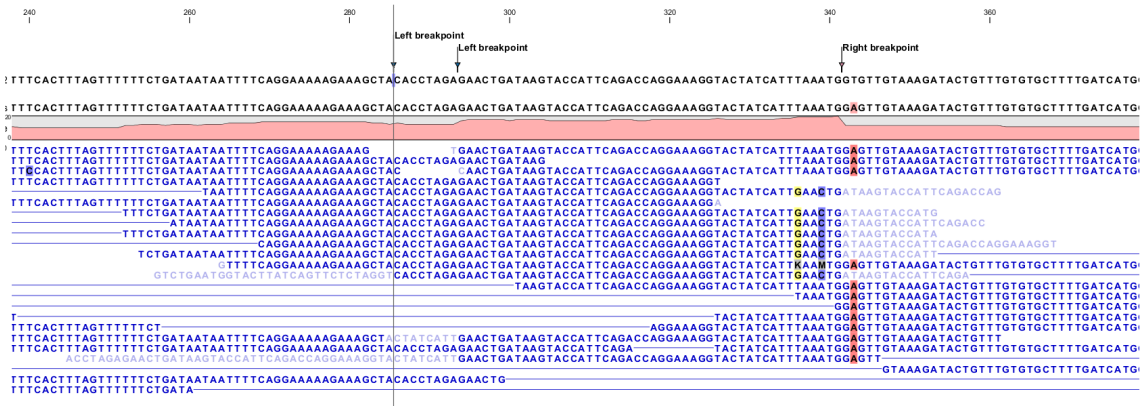


Figure 22.44: *Example of a read mapping containing unaligned ends with three unaligned end signatures.*

3. Calculate the coverage in each position of 'valid' reads with a starting right unaligned end, r_i (of minimum consensus length 3bp).

We then use the observed fractions of 'Left unaligned ends' ($\sum_i l_i / \sum_i c_i$) and 'Right unaligned ends' ($\sum_i r_i / \sum_i c_i$) as frequencies in binomial distributions of 'Left unaligned end' and 'Right unaligned end' read fractions. We go through each position in the read mapping and examine it for an excess of left (or right) unaligned end reads: if the probability of obtaining the observed number of left (or right) unaligned ends in a position with the observed coverage, is 'small', a Left breakpoint signature (LB), respectively Right breakpoint signature (RB), is created.

The two user-specified settings 'The P-value threshold' and the 'Maximum number of mismatches' determine which breakpoint signatures the algorithm will detect (see section 22.9.1 and Figure 22.41). The p-value is used as a cutoff in the binomial distributions estimated above: if the probability of obtaining the observed number of left (or right) unaligned ends in a position with the observed coverage, is smaller than the user-specified cut-off, a Left breakpoint signature (LB), respectively Right breakpoint signature (RB), is created. The 'Maximum number of mis-matches' parameter is used to determine which reads are considered 'valid' unaligned end reads. Only reads that have at most this number of mis-matches in their aligned parts are counted. The higher these two values are set, the more breakpoints will be called. The more breakpoints are called, the larger the search space for the Structural variation detection algorithm, and thus the longer the computation time.

In figure 22.44, three unaligned end signatures are shown. The left-most LB signature is called only when the p-value cut-off is chosen high (0.01 as opposed to 0.0001).

22.9.5 The InDels and Structural Variants detection algorithm - Step 2: Creating Structural variant signatures

In the second step of the InDels and Structural Variants detection algorithm the unaligned end 'breakpoint signatures' (identified in step 1) are used to derive 'structural variant signatures'. This is done by:

1. Generating a consensus sequence of the reads with unaligned ends at each identified breakpoint.

2. Mapping the generated consensus sequences against the reference sequence in the regions around *other* identified breakpoints ('cross-mapping').
3. Mapping the generated consensus sequences of breakpoints that are *near each other against each other* ('aligning').
4. Mapping the generated consensus sequences against the reference sequence in the *region around the breakpoint itself* ('self-mapping').
5. Considering the breakpoints whose unaligned end consensus sequences are found to cross map against each other together, and compare their mapping patterns to the set of theoretically expected 'structural variants signatures' (see section 22.9.6).
6. Creating a 'structural variant signature' for each of the groups of breakpoints whose mapping patterns were in accordance with one of the expected 'structural variants signatures'.

A structural variant is called for each of the created 'structural variant signatures'. For each of the groups of breakpoints whose mapping patterns were NOT in accordance with one of the expected 'structural variants signatures', we call a structural variant of type 'complex'.

The steps above require a number of decisions to be made regarding (1) When is the consensus sequence reliable enough to work with?, and (2) When does an unaligned end map well enough that we will call it a match? The algorithm uses a number of hard-coded values when making those decisions. The values are described below.

Algorithmic details

- **Generating a consensus:** The consensus of the unaligned ends is calculated by simple alignment without gaps. Having created the consensus, we exclude the unaligned ends which differ by more than 20% from the consensus, and recalculate the consensus. This prevents 'spuriously' unaligned ends that extend longer than other unaligned ends from impacting the tail of the consensus unaligned end.
- **Mapping of the consensus:**
  - **'Cross mapping':** When mapping the consensus sequences against the reference sequence around other breakpoints we require that:
    - The consensus is at least 16 bp long.
    - The score of the alignment is at least 70% of the maximal possible score of the alignment.
  - **'Aligning':** When aligning the consensus sequences two closely located breakpoints against each other we require that:
    - The breakpoints are within a 100 bp distance of each other.
    - The overlap in the alignment of the consensus sequences is least 4 nucleotides long.
  - **'Self-mapping':** When mapping the consensus sequences of breakpoints against the reference sequence in a region around the breakpoint itself we require that:
    - The consensus is at least 9 bp long.
    - A match is found within 400 bp window of the breakpoint.

* The score of the alignment is at least 90% of the maximal possible score of the alignment of the part of the consensus sequence that does not include the variant allele part.

22.9.6 Theoretically expected structural variant signatures

Different types of structural variants will leave different 'signatures' in terms of the mapping patterns of the unaligned ends. The 'structural variant signatures' of the set of structural variants that are considered by the Indels and Structural variant tool are drawn in figures 22.45, 22.46, 22.47, 22.48, 22.49, 22.50, 22.51, 22.52 and 22.53.

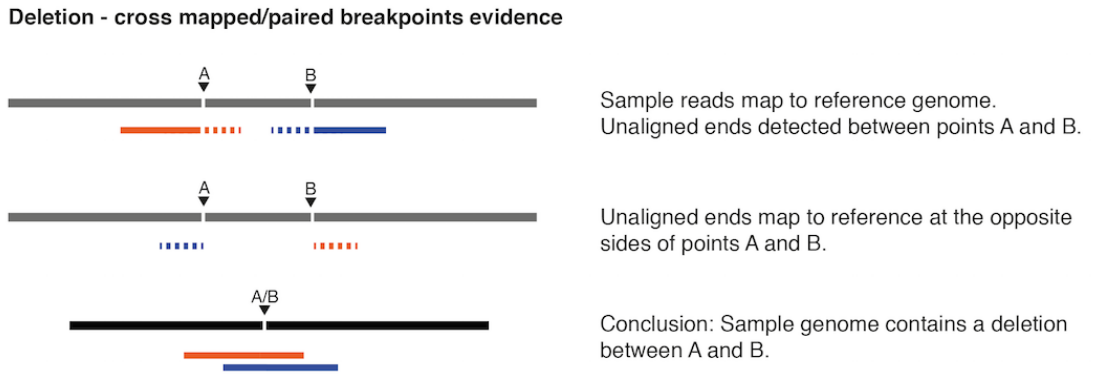


Figure 22.45: *A deletion with cross-mapping breakpoint evidence.*

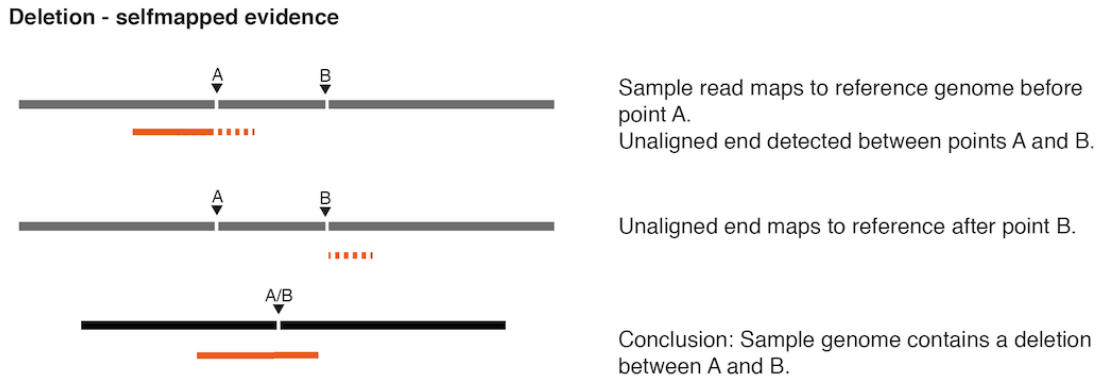


Figure 22.46: *A deletion with selfmapping breakpoint evidence.*

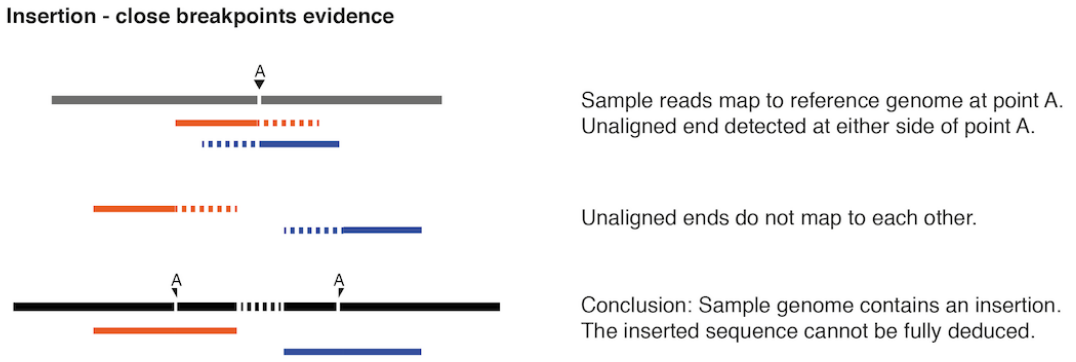


Figure 22.47: *An insertion with close breakpoint evidence.*

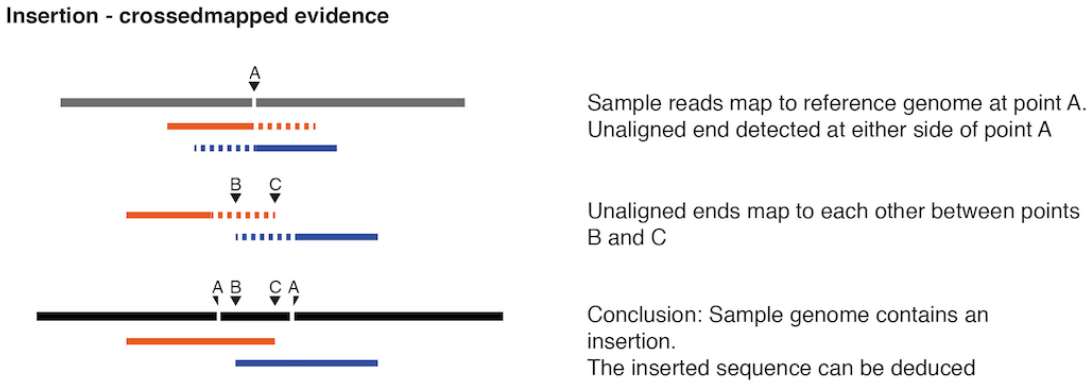


Figure 22.48: *An insertion with cross-mapped breakpoints evidence.*

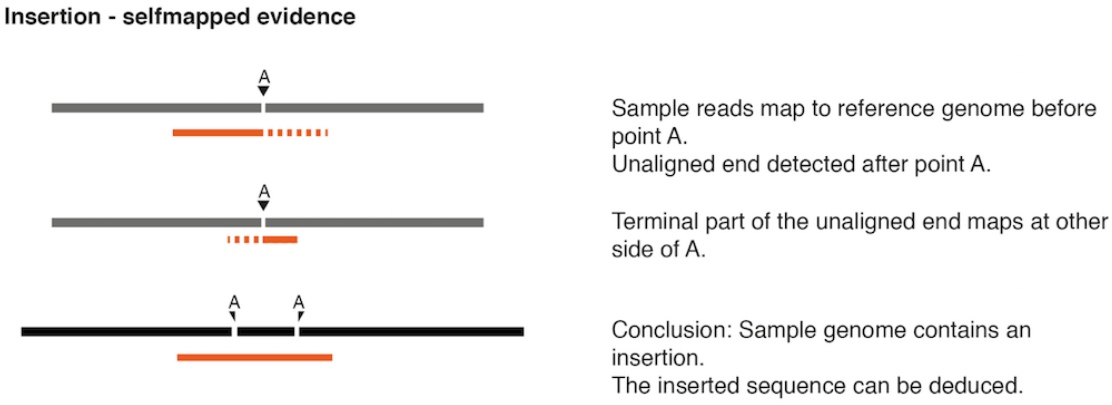


Figure 22.49: *An insertion with selfmapped breakpoint evidence.*

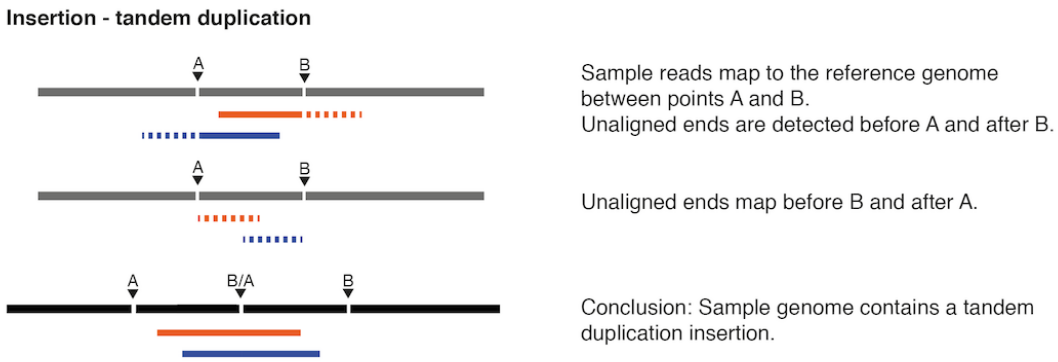


Figure 22.50: *An insertion with breakpoint mapping evidence corresponding to a 'Tandem duplication'.*

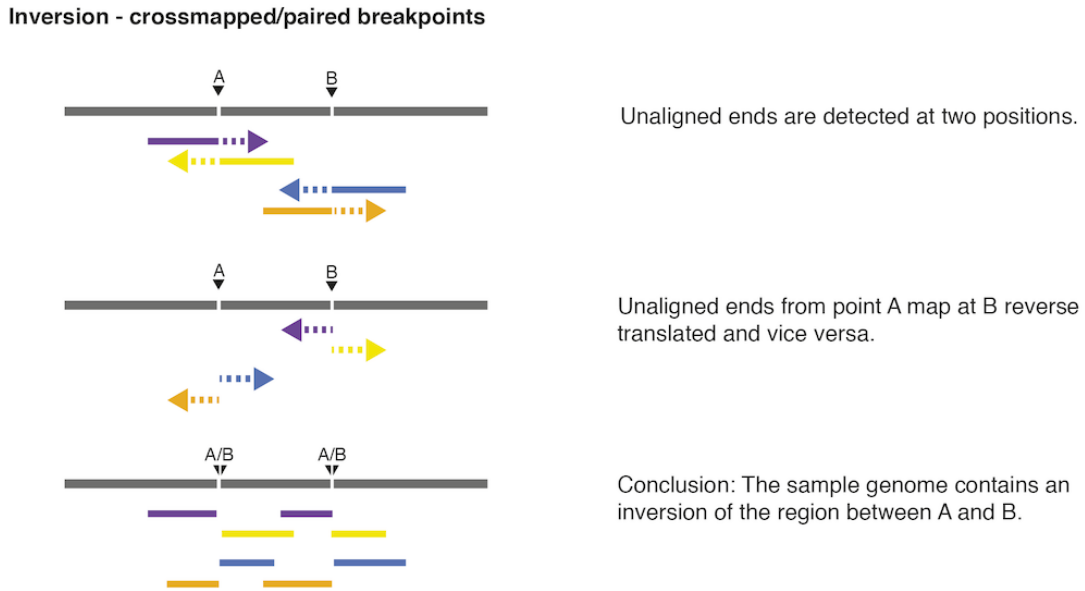


Figure 22.51: *The unaligned end mapping pattern of an inversion.*

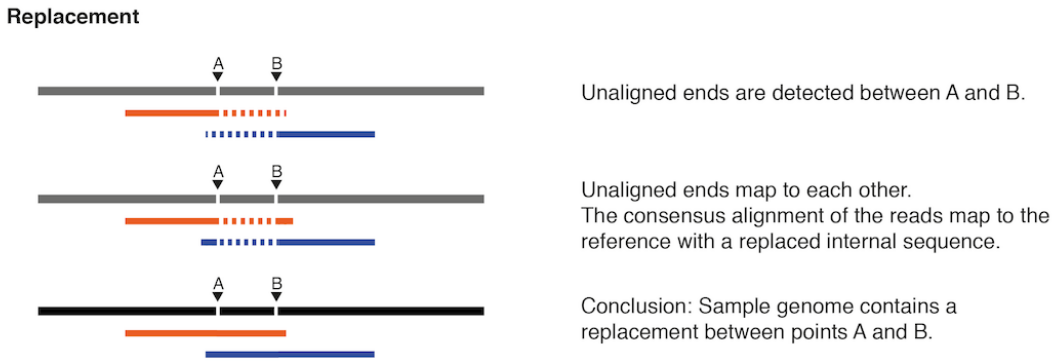


Figure 22.52: *The unaligned end mapping pattern of a replacement.*

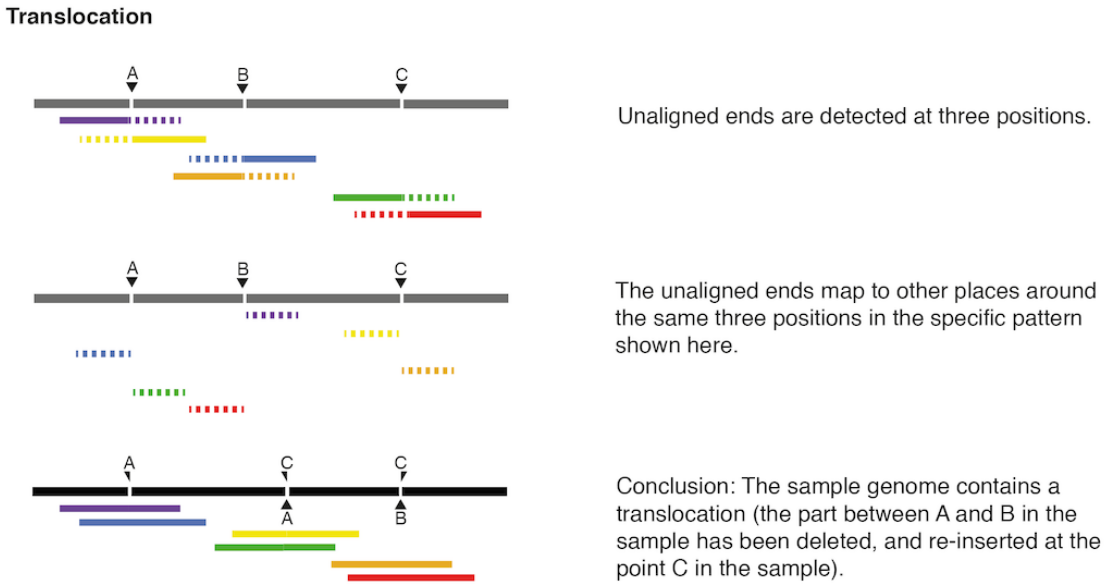


Figure 22.53: *The unaligned end mapping pattern of a translocation.*

22.9.7 How sequence complexity is calculated

The sequence complexity of an unaligned end is calculated as the product of 'the observed vocabulary-usages' divided by 'the maximal possible vocabulary-usages', for word sizes from one to seven. When multiple breakpoints are used to construct a structural variant, the complexity is calculated as the product of the individual sequence complexities of the breakpoints constituting the structural variant.

The observed vocabulary usage for word size, k, for a given sequence is the number of different "words" of size k that exist in that sequence. The maximal possible vocabulary usage for word size k for a given sequence is the maximal number of different words of size k that can possibly be observed in a sequence of a given length. For DNA sequences, the set of all possible letters in such words is four, that is, there are four letters that represent the possible nucleotides: A, C, G and T. The calculation is most easily described using an example.

Consider the sequence CAGTACAG. In this sequence we observe:

- 4 different words of size 1 ('A,', 'C', 'G' and 'T').
- 5 different words of size 2 ('CA', 'AG', 'GT', 'TA' and 'AC') Note that 'CA' and 'AG' are found twice in this sequence.
- 5 different words of size 3 ('CAG', 'AGT', 'GTA', 'TAC' and 'ACA') Note that 'CAG' is found twice in this sequence.
- 5 different words of size 4 ('CAGT', 'AGTA', 'GTAC', 'TACA' and 'ACAG')
- 4 different words of size 5 ('CAGTA', 'AGTAC' , 'GTACA' and 'TACAG')
- 3 different words of size 6 ('CAGTAC', 'AGTACA' and 'GTACAG')
- 2 different words of of size 7 ('CAGTACA' and 'AGTACAG')

Note that we only do the calculations for word sizes up to 7, even when the unaligned end is longer than this.

Now we consider the maximal possible number of words we could observe in a DNA sequence of this length, again restricting our considerations to word lengths of 7.

- Word size of 1: The maximum number of different letters possible here is 4, the single characters, A, G, C and T. There are 8 positions in our example sequence, but there are only 4 possible unique nucleotides.
- Word size of 2: The maximum number of different words possible here is 7. For DNA generally, there is a total of 16 different dinucleotides (4*4). For a sequence of length 8, we can have a total of 7 dinucleotides, so with 16 possibilities, the dinucleotides at each of our 7 positions could be unique.
- Word size of 3: The maximum number of different words possible here is 6. For DNA generally, there is a total of 64 different dinucleotides (4*4*4). For a sequence of length 8, we can have a total of 6 trinucleotides, so with 64 possibilities, the trinucleotides at each of our 6 positions could be unique.

- Word size of 4: The maximum number of different words possible here is 5. For DNA generally, there is a total of 256 different dinucleotides (4*4*4*4). For a sequence of length 8, we can have a total of 5 quatronucleotides, so with 256 possibilities, the quatronucleotides at each of our 5 positions could be unique.

We then continue, using the logic above, to calculate a maximum possible number of words for a word size of 5 being 4, a maximum possible number of words for a word size of 6 being 3, and a maximum possible number of words for a word size of 7 being 2.

Now we can compute the complexity for this 7 nucleotide sequence by taking the number of different words we observe for each word length from 1 to 7 nucleotides and dividing them by the maximum possible number of words for each word length from 1 to 7. Here that gives us:

(4/4)(5/7)(5/6)(5/5)(4/4)(3/3)(2/2) = 0.595

As an extreme example of a sequence of low complexity, consider the 7 base sequence AAAAAAA. Here, we would get the complexity:

(1/4)(1/6)(1/5)(1/4)(1/3)(1/2)(1/1) = 0.000347

22.10 Copy Number Variant Detection

The Copy Number Variant Detection tool is designed to detect copy number variations (CNVs) from targeted resequencing experiments.

The tool takes read mappings and target regions as input, and produces amplification and deletion annotations. The annotations are generated by a 'depth-of-coverage' method, where the target-level coverages of the case and the controls are compared in a statistical framework using a model based on 'selected' targets. Note that to be 'selected', a target has to have a coverage higher than the specified coverage cutoff AND must be found on a a chromosome that was not identified as a coverage outlier in the chromosomal analysis step. If fewer than 50 'selected' targets are found suitable for setting up the statistical models, the CNV tool will terminate prematurely.

The algorithm implemented in the Copy Number Variant Detection tool is inspired by the following papers:

- **Li et al.**, *CONTRA: copy number analysis for targeted resequencing*, Bioinformatics. 2012, 28(10):1307-1313 [Li et al., 2012].
- **Niu and Zhang**, *The screening and ranking algorithm to detect DNA copy number variations*, Ann Appl Stat. 2012, 6(3): 1306-1326 [Niu and Zhang, 2012].

The Copy Number Variant Detection tool identifies CNVs regions where the normalized coverage is statistically significantly different from the controls.

The algorithm carries out the analysis in several steps.

1. Base-level coverages are analyzed for all samples, and a robust coverage baseline is generated using the control samples.

2. Chromosome-level coverage analysis is carried out on the case sample, and any chromosomes with unexpectedly high or low coverages are identified.
3. Sample coverages are normalized, and a global, target-level statistical model is set up for the variation in fold-change as a function of coverage in the baseline.
4. Each chromosome is segmented into regions of similar fold-changes.
5. The expected fold-change variation in region is determined using the statistical model for target-level coverages. Region-level CNVs are identified as the regions with fold-changes significantly different from 1.0.
6. If chosen in the parameter steps, gene-level CNV calls are also produced.

22.10.1 Running the Copy Number Variant Detection tool

Algorithm and parameter description

To start the Copy Number Variant Detection tool, click:

Toolbox | **Resequencing Analysis (**  **)** | **Copy Number Variant Detection**

Select the case read mapping and click **Next**. You are now presented with choices regarding the data to use in the CNV prediction method, as shown in figure 22.54.

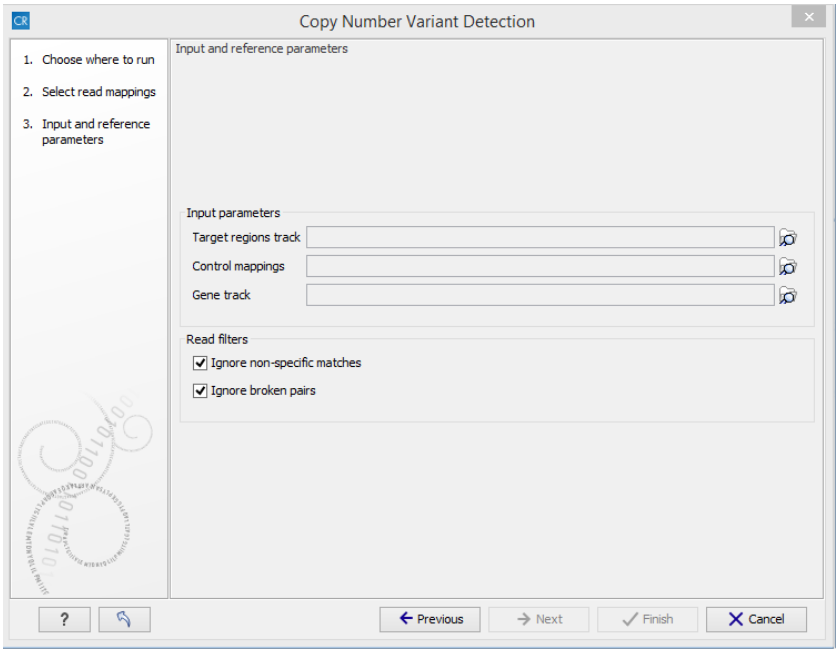


Figure 22.54: *The first step of the CNV detection tool.*

Target regions track An annotation track containing the regions targeted in the experiment must be chosen. This track must not contain overlapping regions, or regions made up of several intervals, because the algorithm is designed to operate on simple genomic regions.

Control mappings You must specify one or more read mappings, which will be used to create a baseline by the algorithm. For the best results, the controls should be matched with

respect to the most important experimental parameters, such as gender and technology. If using non-matched controls, the CNVs reported by the algorithm may be less accurate.

Gene track Optional: If you wish, you can provide a gene track, which will be used to produce gene-level output as well as CNV-level output.

Ignore non-specific matches If checked, the algorithm will ignore any non-specifically mapped reads when counting the coverage in the targeted positions. Note: If you are interested in predicting CNVs in repetitive regions, this box should be unchecked.

Ignore broken pairs If checked, the algorithm will ignore any broken paired reads when counting the coverage in the targeted positions.

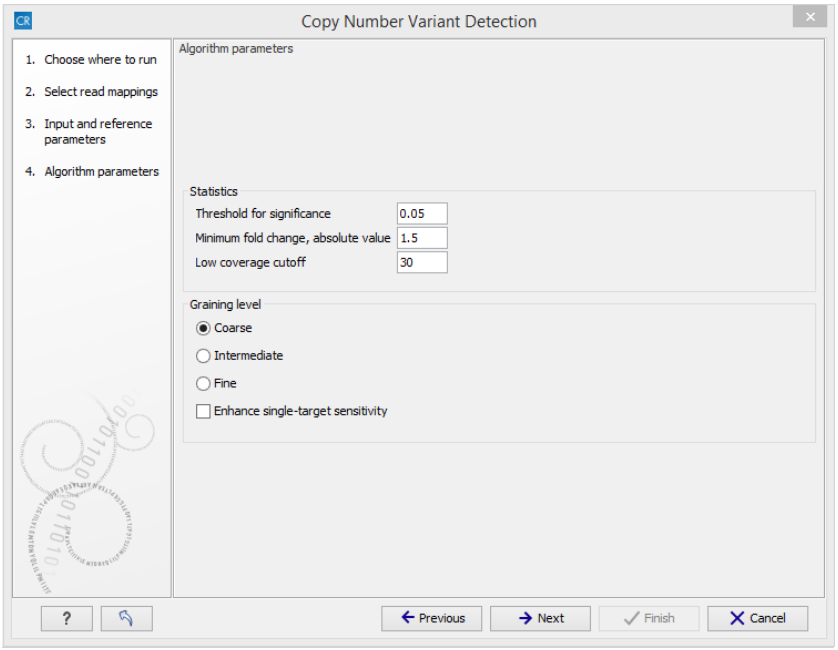


Figure 22.55: *The second step of the CNV detection tool*

Click **Next** to set the parameters related to the target-level and region-level CNV detection, as shown in as shown in figure 22.55.

Threshold for significance P-values lower than the threshold for significance will be considered "significant". The higher you set this value, the more CNVs will be predicted.

Minimum fold change, absolute value You must specify the minimum fold change for a CNV call. If the absolute value of the fold change of a CNV is less than the value specified in this parameter, then the CNV will be filtered from the results, even if it is otherwise statistically significant. For example, if a minimum fold-change of 1.5 is chosen, then the adjusted coverage of the CNV in the case sample must be either 1.5 times higher or 1.5 times lower than the coverage in the baseline, for it to pass the filtering step. If you do not want to filter on the fold-change, enter 0.0 in this field. Note: If your sample purity is less than 100%, it is necessary to take that into account when you adjust the fold-change cutoff. This is described in more detail in section 22.10.1.

Low coverage cutoff If the average coverage of a target is below this value, it will be considered "low coverage" and it will not be used to set up the statistical models, and p-values will not be calculated for it in the target-level CNV prediction.

Graining level The graining level is used for the region-level CNV prediction. Coarser graining levels produce longer CNV calls and less noise, and the algorithm will run faster. However, smaller CNVs consisting of only a few targets may be missed at a coarser graining level.

- Coarse: prefers CNVs consisting of many targets. The algorithm is most sensitive to CNVs spanning over 10 targets. This is the recommended setting if you expect large-scale deletions or insertions, and want a minimal false positive rate.
- Intermediate: prefers CNVs consisting of an intermediate number of targets. The algorithm is most sensitive to CNVs spanning 5 or more targets. This is the recommended setting if you expect CNVs of intermediate size.
- Fine: prefers CNVs consisting of fewer targets. The algorithm is most sensitive to CNVs spanning 3 or more targets. This is the recommended setting if you want to detect CNVs that span just a few targets, but the false positive rate may be increased.

Note: The CNV sizes listed above are meant as general guidelines, and are not to be interpreted as hard rules. Finer graining levels will produce larger CNVs when the signals for this are sufficiently clear in the data. Similarly, the coarser graining levels will also be able to predict shorter CNVs under some circumstances, although with a lower sensitivity.

Enhance single-target sensitivity All of the graining levels assume that a CNV spans more than one target. If you are also interested in very small CNVs that affect down to a single target in your data, check the 'Enhance single-target sensitivity' box. This will increase the sensitivity of detection of very small CNVs, and has the greatest effect in the case of the coarser graining levels. Note however that these small CNV calls are much more likely to be false positives. If this box is unchecked, only larger CNVs supported by several targets will be reported, and the false positive rate will be lower.

Clicking **Next**, you are presented with options about the results (see figure 22.56). In this step, you can choose to create an algorithm report by checking the **Create algorithm report** box. Furthermore, you can choose to output results for every target in your input, by checking the **Create target-level CNV track** box.

When finished with the settings, click **Next** to start the algorithm.

How to set the fold-change cutoff when the sample purity is not 100%

Given a sample purity of $X\%$, and a desired detection level (absolute value of fold-change in 100% pure sample) of T, the following formula gives the required fold-change cutoff:

$$\text{cutoff} = \frac{X\%}{100\%} \times T + (1 - \frac{X\%}{100\%}) \tag{22.6}$$

For example, if the sample purity is 40%, and you want to detect 6-fold amplifications or deletions (e.g. 12 copies instead of 2, or 2 copies instead of 12), then the cutoff should be:

$$\text{cutoff} = \frac{40\%}{100\%} \times 6 + (1 - \frac{40\%}{100\%}) = 3.0. \tag{22.7}$$

Figures 22.57 shows the required fold-change cutoffs in order to detect a particular degree of amplification/deletion at different sample purities. Figure 22.58 zooms in for low-level amplifications and deletions.

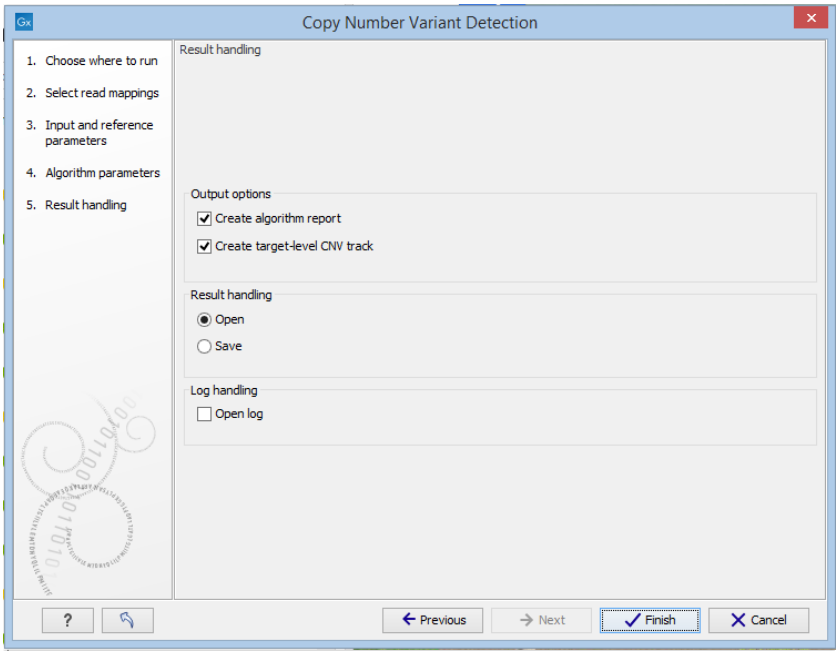


Figure 22.56: *Specifying whether an algorithm report and a target-level CNV track should be created.*

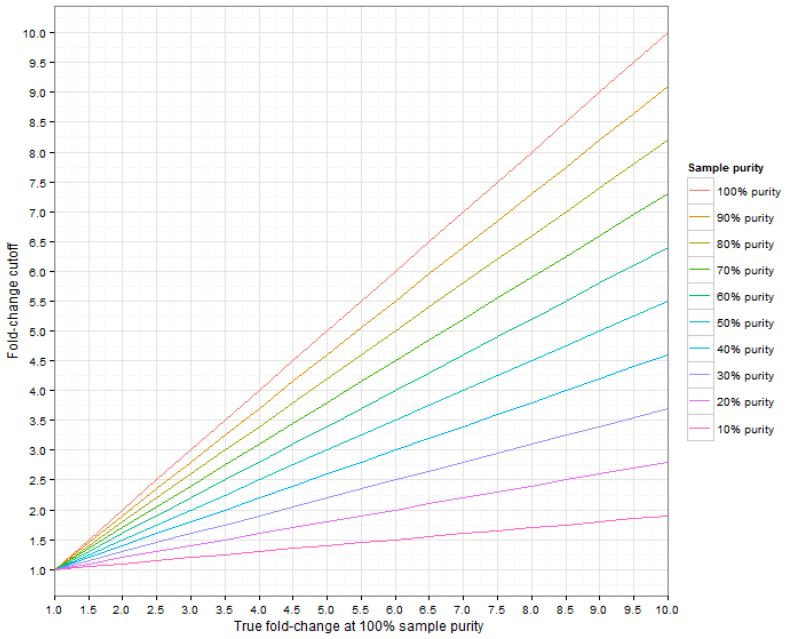


Figure 22.57: *The required fold-change cutoff to detect amplifications and deletions of different magnitudes, as a function of sample purity.*

The Copy Number Variant Detection tool calls CNVs that are both global outliers on the target-level, and locally consistent on the region-level. The tool produces several outputs, which are described below.

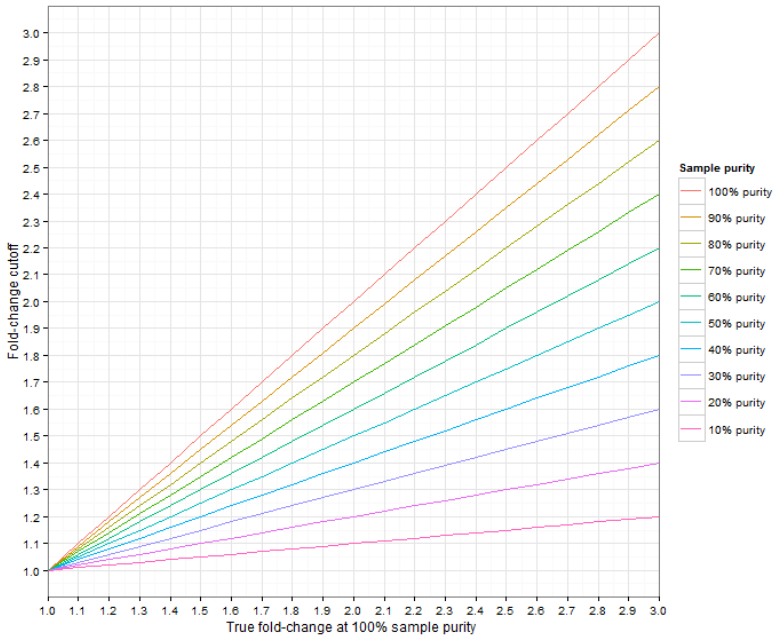


Figure 22.58: *Low-level amplifications and deletions: The required fold-change cutoff to detect amplifications and deletions of different magnitudes, as a function of sample purity.*

22.10.2 Region-level CNV track (Region CNVs)

The algorithm will produce a region-level annotation track, which contains the CNV regions detected by the algorithm. Every annotation in this track joins one more more targets from the input target track, to produce contiguous CNVs. Each CNV in the region-level tracks is characterized in terms of the following properties:

Minimum CNV length: The minimum CNV length is the length of the region-level CNV annotation. This number should be interpreted as the lowest bound for the size of the CNV. The "true" CNV can extend into the adjacent genomic regions that have not be targeted.

P-value: The p-value corresponds to the probability that an observation identical to the CNV, or even more of an outlier, would occur by chance under the null hypothesis. The null hypothesis is that of no CNVs in the data. The p-value for a CNV region is calculated by combining the p-values of its constituent targets (discarding any low-coverage targets).

Fold-change (adjusted): The fold-change of the *adjusted* case coverage compared to the baseline. Negative fold-changes indicate deletions, and positive fold-changes indicate amplifications. A fold-change of 1.0 (or -1.0) represents identical coverages. The fold-changes are adjusted for statistical differences between targets with different sequencing depths. Note: if your sample purity is less than 100%, you need to take that into account when interpreting the fold-change values. This is described in more detail in section 22.10.2.

Consequence: The consequence classifies statistically significant CNVs as "Gain" or "Loss".

Number of targets: The total number of targets forming the (minimal) CNV region.

Targets: A list of the names of the targets forming the (minimal) CNV region. Note however that the list is truncated to 100 characters. If you want to see all the targets that constitute the CNV region, you can use the target-level output (section 22.10.3).

Comments: The comments can include useful information for interpreting individual CNV calls. The possible comments are:

1. Small region: If a region only consists of 1 target, it is classified as a 'small region'. The p-value of this region is therefore based on evidence from just one target, and may be less accurate than p-values for larger regions.
2. Disproportionate chromosome coverage: If a region is found on a chromosome that was determined to have disproportionate coverage, this will be noted in the comments. This means that the targets constituting this region were not used to set up the statistical models. Furthermore, the size and fold-change value of this CNV region may explain why the chromosome was detected to have disproportionate coverage.
3. Low coverage: If all targets inside a region had low-coverage, then the region will be classified as a 'low-coverage' region, and will be given a p-value of 1.0. You will only see these regions in the results if you set the significance cutoff to 1.0.

These properties can be found in separate columns when viewing the tracks in table view.

Note: The region-level calls do not guarantee that a single, larger CNV will always be called in just one CNV region. This is because adjacent region-level CNV calls are not joined into a single region if their average fold-changes are sufficiently different. For example, if a 2-fold gain is detected in a region and a 3-fold gain is detected in an immediately adjacent region of equal size, then these may appear in the results as two separate CNVs, or one single CNV with a 2.5-fold gain, depending on your chosen graining level, and the fold-changes observed in the rest of the data.

How to interpret fold-changes when the sample purity is not 100%

If your sample purity is less than 100%, it is necessary to take that into account when interpreting the fold-change values. Given a sample purity of X%, and an observed fold-change of F, following formula gives the actual fold-change that would be seen if the sample were 100% pure:

$$\text{fold-change in 100\% pure sample} = \frac{F-1}{X/100\%} + 1 \tag{22.8}$$

For example, if the sample purity is 40%, and you have observed a fold-change of 3, then the fold-change in the 100% pure sample would have been:

$$\text{fold-change in 100\% pure sample} = \frac{3.0-1}{40\%/100\%} + 1 = 6.0. \tag{22.9}$$

Figures 22.59 shows the 'true' fold changes for different observed fold-changes at different sample purities. Figure 22.60 zooms in for low-level amplifications and deletions.

22.10.3 Target-level CNV track (Target CNVs)

The algorithm will produce a target-level CNV track, if you've chosen to create one when running the algorithm. The target-level CNV track is an annotation track, containing one annotation for every target in the input data. Inspection of the target-level CNV track can give you additional information about both the CNVs called in the region-level results, and those regions that have

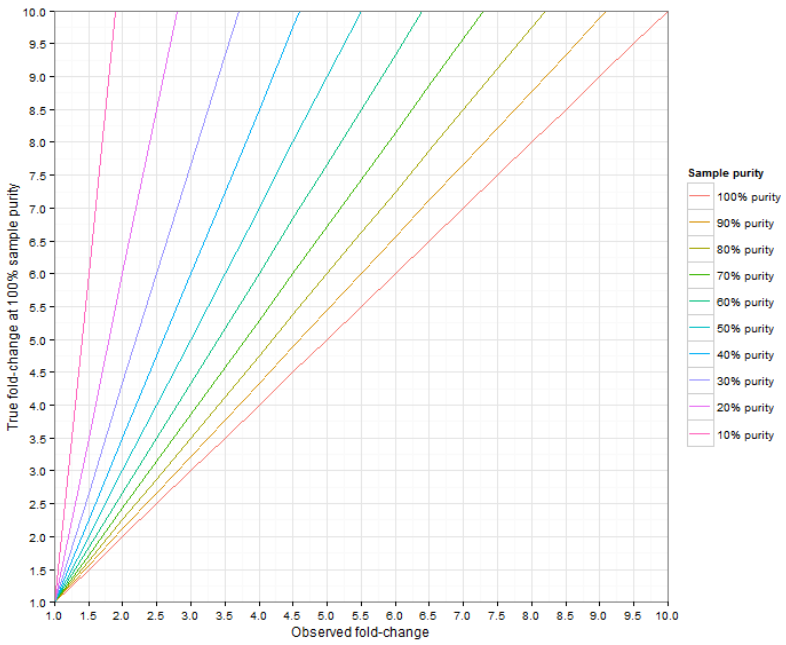


Figure 22.59: *The true fold-change in the 100% pure sample, for different observed fold-changes, as a function of sample purity.*

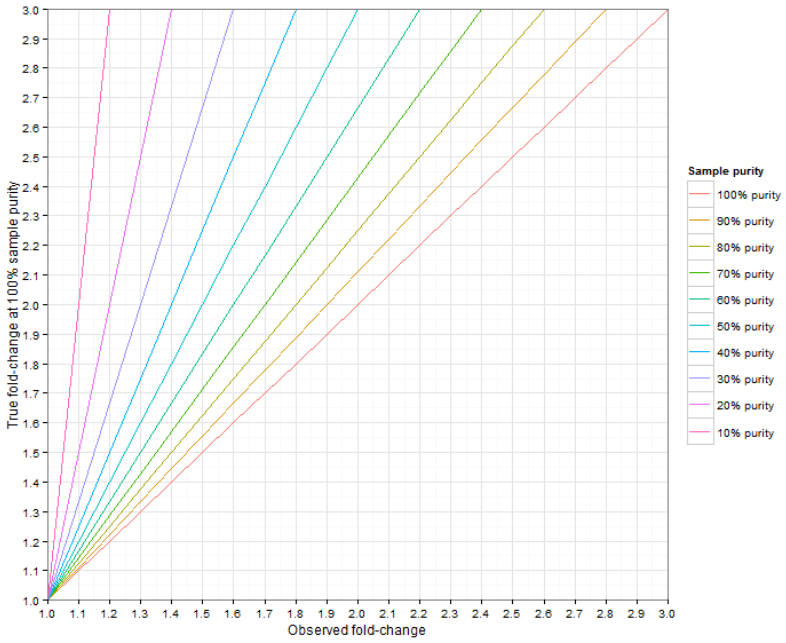


Figure 22.60: *Low-level amplifications and deletions: the true fold-change in the 100% pure sample, for different observed fold-changes, as a function of sample purity.*

not been called. Note that a "statistically relevant" target is one that has a coverage higher than the specified coverage cutoff, AND is found on a a chromosome that was not identified as a coverage outlier in the chromosomal analysis step. The sample is not considered covered enough for statistical purposes if you have fewer than 49 targets that can be used to do the statistics.

Each target is annotated with the following information:

Target number: Targets are numbered in the order in which they occur in the genome. This information is used by the results report (see section 22.10.5).

Case coverage: The normalized coverage of the target in the case sample.

Baseline coverage: The normalized coverage of the target in the baseline.

Length: The length of the target region.

P-value: The p-value corresponds to the probability that an observation identical to the CNV, or even more of an outlier, would occur by chance under the null hypothesis. The null hypothesis is that of no CNVs in the data. The p-value in the target-level output reflects the global evidence for a CNV at that particular target. The target-level p-values are combined to produce the region-level p-values in the region-level CNV output.

FDR-corrected p-value: The FDR-corrected p-values correct for false positives arising from carrying out a very high number of statistical tests. The FDR-corrected p-value will, therefore, always be larger than the uncorrected p-value.

Fold-change (raw): The fold-change of the normalized case coverage compared to the normalized baseline coverage. The normalization corrects for the effects of different library sizes between the different samples. Negative fold-changes indicate deletions, and positive fold-changes indicate amplifications. A fold-change of 1.0 represents identical coverages.

Fold-change (adjusted): As observed by Li et al (2012, [Li et al., 2012]), the fold-changes (raw) depend on the coverage. Therefore, the fold-changes have to be adjusted for statistical differences between targets with different sequencing depths, before the statistical tests are carried out. The results of this adjustment are found in the "Fold-change (adjusted)" column. Note that sometimes, this will mean that a change that appears to be an amplification in the "raw" fold-change column may appear to be a deletion in the "adjusted" fold-change column, or vice versa. This is simply because for a given coverage level, the raw fold-changes were skewed towards amplifications (or deletions), and this effect was corrected in the adjustment. Note: if your sample purity is less than 100%, you need to take that into account when interpreting the fold-change values. This is described in more detail in section 22.10.2.

Region (joined targets): The region to which this target was classified to belong. The region may or may not have been predicted to be a CNV.

Regional fold-change: The adjusted fold-change of the region to which this target belongs. This fold-change value is computed from all targets constituting the region.

Regional p-value: The p-value of the region to which this target belongs. This is the p-value calculated from combining the p-values of the individual targets inside the region.

Regional consequence: If the target is included in a CNV region, this column will show "Gain" or "Loss", depending on the direction of change detected for the region. Note, however, that the change detected for the region may be inconsistent with the fold-change for a single target in the region. The reason for this is typically statistical noise at the single target.

Regional effect size: The effect size of a target-level CNV reflects the magnitude of the observed fold-change of the CNV region in which the target was found. The effect size of a CNV is classified into the following categories: "Strong" or "Weak". The effect size is "Strong" if

the fold-change exceeds the fold-change cutoff specified in the parameter steps. Otherwise, the effect size will be "Weak". Note, however, that "Weak" CNV calls will be filtered from the region-level output.

Comments: The comments can include useful information for interpreting the CNV calls. Possible comments in the target-level output are:

1. Low coverage target: If the target had a coverage under the specified coverage cutoff, it will be classified as low-coverage. Low-coverage targets were not used in calculating the statistical models, and will not have p-values.
2. Disproportionate chromosome coverage: If the target occurred on a chromosome that was detected to have disproportionate coverage. In this case, the target was not used to set up the statistical models.
3. Atypical fold-change in region: If there is a discrepancy between the direction of fold-change detected for the target and the direction of fold-change detected for the region, then the fold-change of the target is "atypical" compared to the region. This is usually due to statistical noise, and the regional fold-change is likely to be more accurate in the interpretation, especially for large regions.

22.10.4 Gene-level annotation track (Gene CNVs)

If you have specified a gene track in the input parameters, you will get a gene-level CNV track as well. The gene-level CNV track is an annotation track, which is obtained by intersecting the region-level CNV track with the gene track in the input (ignoring any genes that do not overlap with the targets). Note that a single CNV may be reported several times in different genes, and a single gene may also be reported several times, if it is affected by more than one CNV. In addition to the annotations on the gene track supplied in the input parameters, the gene-level CNV track contains the following annotation columns:

Region length: The length of the actual annotation. That is, the length of the CNV region intersected with the gene.

CNV region: The *entire* CNV region affecting this gene (and possibly other genes).

CNV region length: The length of the *entire* CNV region affecting this gene (and possibly other genes).

Consequence: The consequence classifies statistically significant CNVs as "Gain" or "Loss".

Fold-change (adjusted): The adjusted fold-change of the *entire* CNV region affecting this gene (and possibly other genes).

P-value: The p-value of the *entire* CNV region affecting this gene (and possibly other genes).

Number of targets: The total number of targets forming the *entire* CNV region affecting this gene (and possibly other genes).

Comments: If the CNV region affecting this gene had any comments (as described in section 22.10.2), this will be present in the gene-level results as well.

Targets: A list of the names of the targets forming the (minimal) CNV region forming the *entire* CNV region affecting this gene (and possibly other genes). Note however that the list is truncated to 100 characters. If you want to know the full list of targets inside the CNV region, you can use the target-level output track.

22.10.5 CNV results report

The report contains information about the results of the Copy Number Variant Detection tool.

Normalization The Normalization section gives information about the sample-level and chromosome-level coverages. Any chromosomes with disproportionate coverages are noted; targets on these chromosomes were ignored when setting up the statistical models.

Target-level log2-ratios The target-level coverage log-ratios are presented as a graph. An example is shown in figure 22.61. On the horizontal axis, the targets are placed in the order in which they appear in the genome. On the vertical axis, the adjusted coverage log-ratio of each target is plotted. The black line represents the actually observed mean adjusted log-ratio of coverage for each target. The cyan and red lines represent the 95% confidence intervals of the expected mean adjusted log-ratios of coverages, based on the statistical model. Chromosome boundaries are indicated as vertical lines.

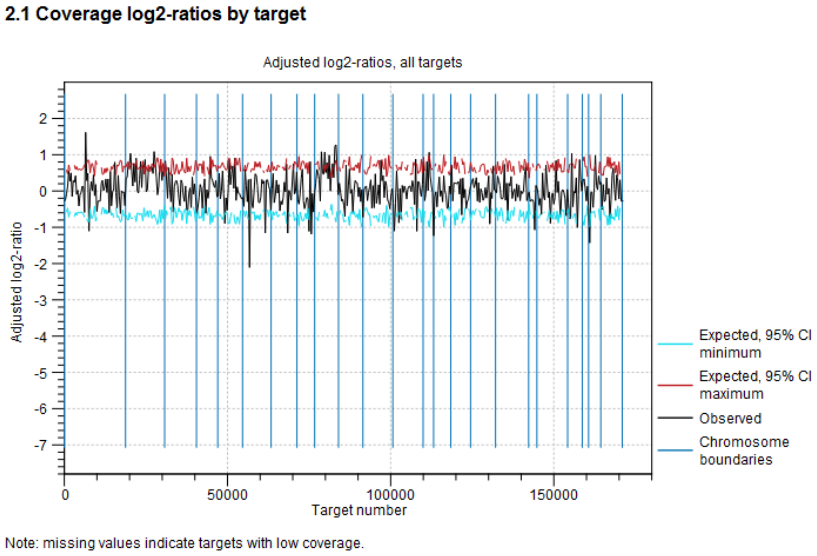


Figure 22.61: *An example graph showing the mean adjusted log-ratios of coverages in the report produced by the Copy Number Variant Detection tool. In this example, the second and ninth chromosomes are amplified, and the log-ratios of coverages of targets on these chromosome are significantly higher than for targets on other chromosomes. The black line in these regions is outside the boundaries defined by the cyan and red lines.*

CNV statistics The last section in the report provides some information about the number of CNVs called in the region-level prediction results. The number of uncalled or filtered regions are also shown.

22.10.6 CNV algorithm report

If you have chosen to produce an algorithm report in the output handling step of the wizard, an algorithm report will also be produced. This contains information about the statistical models of

the algorithm, and can be used to evaluate how well the assumptions of the model were fulfilled. We will now present the different sections of this report.

Normalization and chromosome analysis

This section of the report is related to the first step of the Copy Number Variant Detection tool, where the chromosome-level coverages are analyzed to detect any outliers. The total coverages of the case chromosomes are plotted against the total coverages of the baseline, and the detected outliers are indicated. Chromosome coverages identified as disproportionate are marked with red crosses (see figure 22.62).

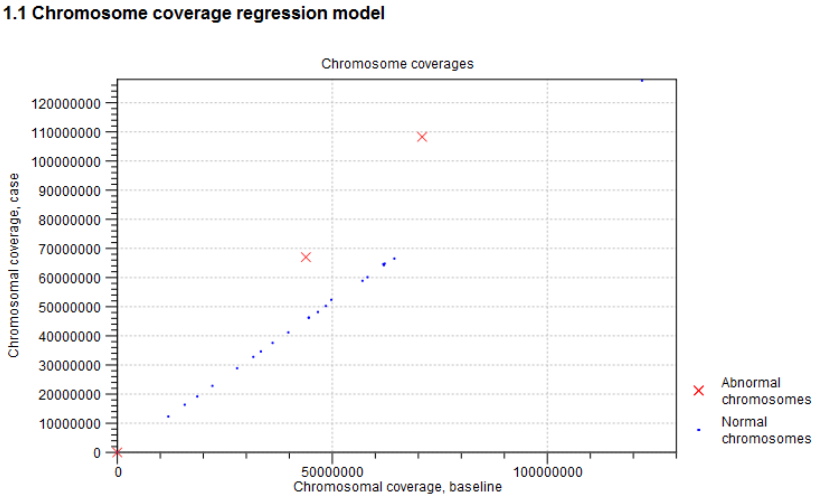


Figure 22.62: *An example graph showing the coverages of the chromosomes in the case versus the baseline. In this example, three chromosomes are marked as abnormal. Two of these chromosomes are significantly amplified, and log-ratios of coverages of many targets on these chromosome are significantly higher than for targets on other chromosomes. The third outlier chromosome had zero coverage in both the case and the baseline.*

The graph is followed by a table, where the detailed chromosome coverages are shown after normalization. Chromosomes with disproportionate coverage and chromosomes without any targets are marked in the 'Comment' column. These chromosomes are the ones marked with red crosses in the graph in section 1.1. of the algorithm report, and these chromosomes were not used in the coverage normalization step.

Prediction of target-level CNVs

This section of the algorithm report gives information about the statistical models used to predict target-level CNVs.

Adjustment of log2-ratios The first two graphs in this section are related to the adjustment of the log-ratios of coverages as a function of log-coverage. The log-ratio of coverages for targets depends on the level of coverage of the target, as observed by Li et al. (Bioinformatics, 2012), who also proposed that a linear correction should be applied [Li et al., 2012]. In the first of the two graphs, the non-adjusted log-ratios of target coverages are plotted against the log-coverage of the targets. In the second graph, the mean log-ratios are plotted after adjustment (figure 22.63). If the model fits the data, we expect to see that the adjusted mean log-ratios are centered around 0 for all log-coverages, and the variation decreases with increasing log-coverage.

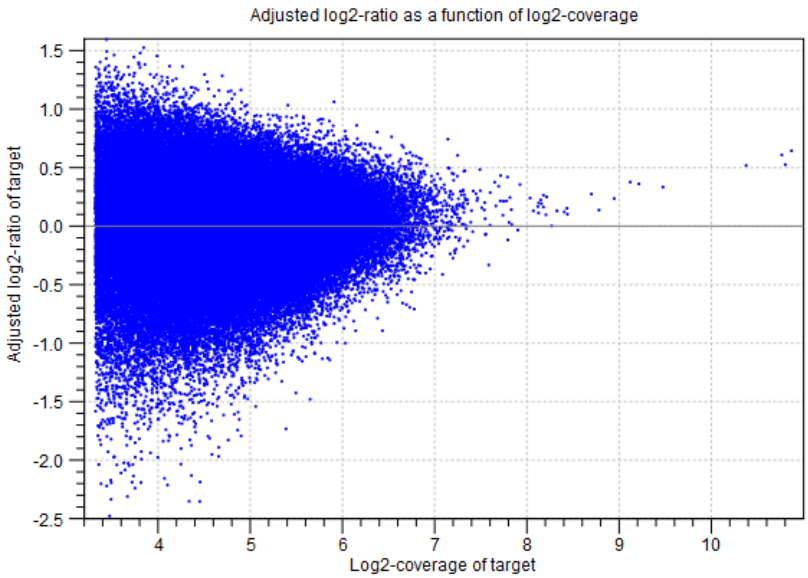


Figure 22.63: *An example graph showing the mean adjusted log-ratios of coverages plotted against the log-coverages of targets, in the algorithm report of the Copy Number Variation Detection tool. Here, the adjusted mean log-ratios are centered around 0.0 for most coverages, and the variation decreases with increasing log-coverage. This indicates a good fit of the model. However, at very high coverages, the adjusted log-ratios are centered higher than 0.0, which indicates that for these coverages, the model is not a perfect fit. But only very few targets are affected by this, as the points are very sparse at these high coverage levels.*

Statistical model for adjusted log2-ratios In this section of the algorithm report, you can see how well the algorithm was able to model the statistical variation in the log-ratios of coverages. An example is shown in figure 22.64). A good fit of the model to the data points indicates that the variance has been modeled accurately.

To make the points more visible, double-click the figure, to open it in a separate editor. Here, you can select how to visualize the data points and the fitted model. For example, you can choose to highlight the data points in the sidepanel:

MA Plot Settings | **Dot properties** | **Dot type** | "Dot"

Distribution of adjusted log2-ratios in bins One of the assumptions of the statistical model used by the CNV detection tool is that the coverage log-ratios of targets are normally distributed with a mean of zero, and the variance only depends on the log-coverage of each target in the baseline. The bar charts in this section of the algorithm report show how well this assumption of the model fits the data. An example is shown in figure 22.65). A good fit of the model to the data points indicates that the variance has been modeled accurately.

Prediction of region-level CNVs

The final section of the algorithm report is related to the region-level CNV prediction. In this part of the algorithm, the chromosomes are segmented into regions of similar adjusted mean log-ratios. More segments lead to a reduced variance per segment; in the extreme, where every target forms its own segment, the variance is zero. However, more segments also mean that the model contains more free parameters, and is therefore potentially over-fitted. A value known as

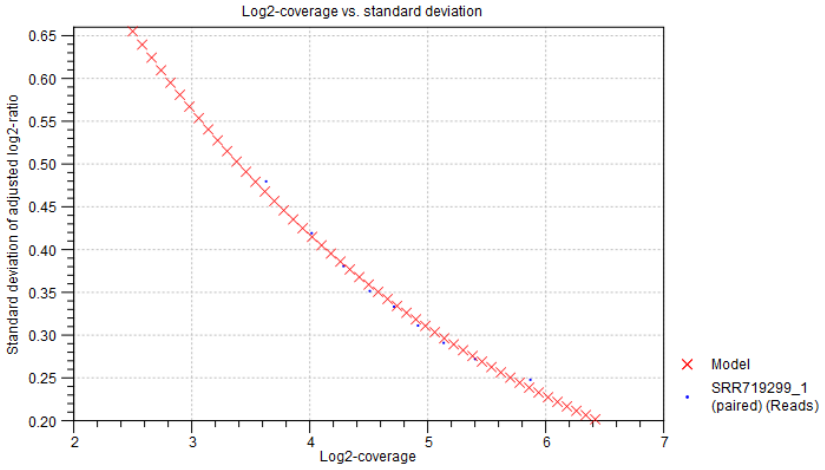


Figure 22.64: *An example graph showing how the variance in the target-level mean log-ratios was modeled in the algorithm report of the Copy Number Variation Detection tool. Here, the data points are very close to the fitted model, indicating a good fit of the model to the data.*

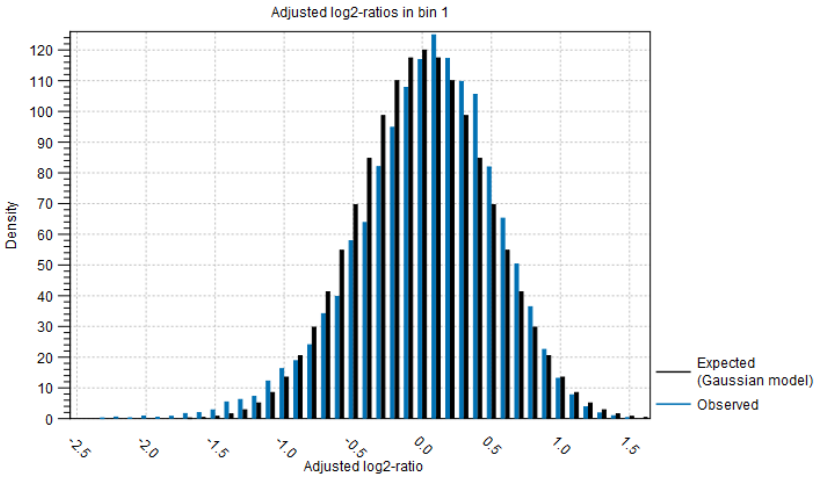


Figure 22.65: *An example bar chart from the algorithm report of the Copy Number Variation Detection tool, showing how well the normal distribution assumption was fulfilled by the adjusted coverage log-ratios. Here, there is a good correspondence between the expected distribution and the observations.*

the Bayesian Information Criterion (BIC) gives an indication of the balance of these two effects, for any potential segmentation of a chromosome. The segmentation process aims to minimize the BIC, producing the best balance of accuracy and overfitting in the final segments.

The segmentation begins by identifying a set of potential breakpoints, known as local maximizers. The number of potential breakpoints at the start of the segmentation is shown in the "# local maximizers at start" column, and the corresponding BIC score is indicated in the "Start BIC" column. Breakpoints are removed strategically one-by-one, and the BIC score is calculated after each removal. When enough breakpoints have been removed for the BIC score to reach its minimum, the final number of breakpoints is shown in the "# local maximizers at end" column, and the corresponding BIC score is indicated in the "End BIC" column. A large reduction in the number of local maximizers indicates that it was possible to join many smaller CNV regions into larger ones.

Note: The segmentation process only produces regions of similar adjusted coverage log-ratios. Each segment is tested afterwards, to identify if it represents a CNV. Therefore, the number of segments shown in this table does not correspond to the number of CNVs actually predicted by the algorithm.

22.11 Coverage analysis

The coverage analysis tool is designed to identify regions in read mappings with unexpectedly low or high coverage. Such regions may be indicative of a deletion or an amplification in the sample relative to the reference. The algorithm fits a Poisson distribution to the observed coverage in the positions of the mapping. This distribution is used as the basis for identifying the regions of 'Low coverage' or 'High coverage'. The user chooses two parameter values in the wizard: (1) a 'Minimum length' and (2) a 'P-value threshold' value. The algorithm inspects the coverage in each of the positions in the read mapping and marks the ones with coverage in the lower or upper tails of the estimated Poisson distribution, using the provided p-value as cut-off. Regions with consecutive positions marked consistently as having low (respectively high) coverage, longer than the user specified 'Minimum length' value are called as 'Low coverage' (respectively 'High coverage') regions.

To run the Coverage analysis tool:

Toolbox | **Resequencing Analysis** ( ) | **Coverage Analysis** ( )

In the first dialog, select a reads track or read mapping and click **Next**. This opens the dialog shown in figure 22.66.

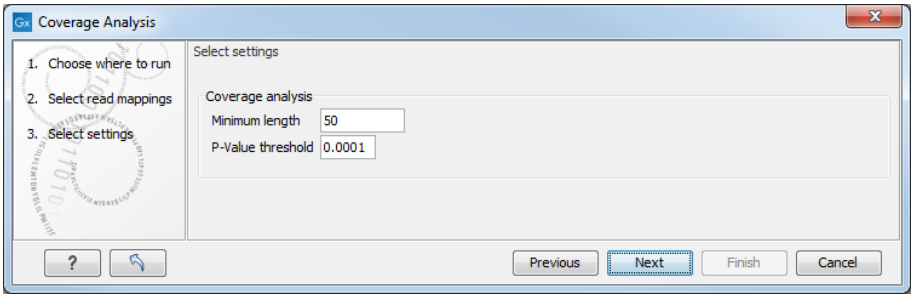


Figure 22.66: *Specify the p-value cutoff.*

Set the p-value and minimum length cutoff. Click **Next** and specify the result handling (figure 22.67).

Selecting the "Create report" will generate the report made of 2 tables (figure 22.68). The first one, called References, lists per chromosome the number of reads, their length, and how many signatures of unexpectedly low or high coverage was found in the mapping. The second table lists on 2 rows low and high coverage signatures found, as well as how many reads were used to calculate these signatures.

Selecting the "Create regions" will generate the annotation track carrying the name of the original file followed by (COV). This file can be visualized as an annotation track or as a table depending on the users choice. The annotation table contains a row for each detected low or high coverage region, with information describing the location, the type and the p-value of the detected region. The p-value of a region is defined as the average of the p-values calculated for each of the positions in the region.

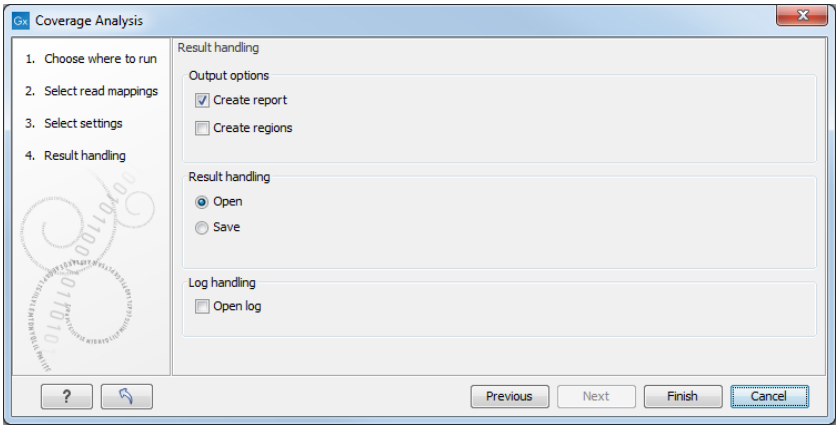


Figure 22.67: *Specify the output.*

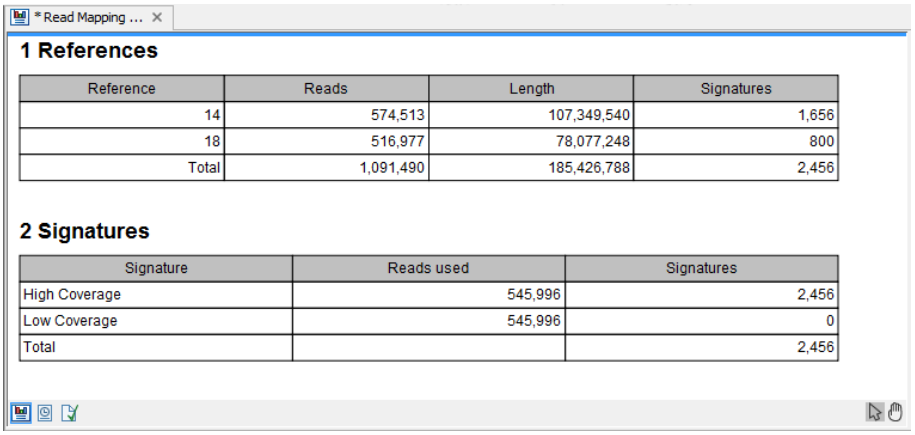


Figure 22.68: *The report output.*

An example of a track output of the Coverage analysis tool is shown in figure 22.69.

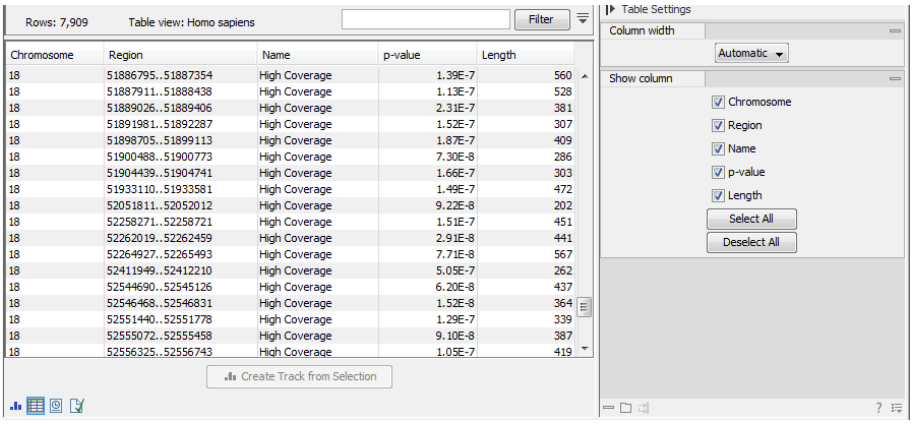


Figure 22.69: *The table output with detailed information on each region.*

The Coverage Analysis table includes the following columns (figure 22.69):

Chromosome The name is taken from the reference sequence used for mapping

Region The start and end position of this region on the reference sequence

Name The type of annotation: high or low coverage

P-Value The calculated significance p-value for this region

Length The length of the region

For the visual inspection and comparison to known gene/transcripts or other kind of annotations, all region are also annotated on the read mapping.

22.12 Variant Detectors - overview

Biomedical Genomics Workbench offers three tools for detecting variants.

- **Fixed Ploidy Variant Detection** () - described in detail in section 22.13
- **Low Frequency Variant Detection** () - described in detail in section 22.14
- **Basic Variant Detection** () - described in detail in section 22.15

They are designed for the analysis of different types of samples and they differ in their underlying assumptions about the data, and hence in their assessments of when there is enough information in the data for a variant to be called. An overview of these differences is given in figure 22.70.

Variant caller	Applications	Data	Variant detected	Comments	Examples of recommended applications
Fixed Ploidy	Germline variants	A sample for which the ploidy can be assumed known	Will detect variants whose representation in the reads is in accordance with the assumed ploidy	Will discard variants whose representation in the reads is likely due to sequencing errors or mapping artefacts	Germline variant calling
Low Frequency	Germline and somatic variants	A sample with unknown/mixed ploidy	Will detect variants whose representation in the reads is in accordance with the presence of a variant in a proportion of the reads	Will discard variants whose representation in the reads is likely due to sequencing errors	Any application where you are looking to detect low frequency (such as non-germline) variants
Basic	Any position that shows at least a specified number and frequency of nucleotide bases that differ from the reference base – irrespective of whether this may be explained by sequencing errors	Any	Will detect any variant observed in the reads	Will call a variant in any position that shows at least a specified number and frequency of nucleotide bases that differ from the reference base – irrespective of whether this may be explained by sequencing errors	exploratory read mapping applications - but not standard variant calling applications

Figure 22.70: *An overview of the variant detection tools.*

To run one of the variant detection tool, go to:

Toolbox | **Resequencing Analysis (**  **)**

and choose the appropriate tool. In the first dialog of each detection tool, you are asked to specify the **reads track** or read mapping to analyze. The user is next asked to set the parameters that are specific for the variant detection tool. The three tools, their assumptions, and the tool-specific parameters are described later in their respective sections.

All variant detection tools will call:

- SNVs - single nucleotide variants
- MNVs - neighboring SNVs, where there is evidence they occur together
- small to medium-sized insertions and deletions - insertions and deletions fully represented within a single read

- replacements - neighboring SNVs and insertions or deletions

22.12.1 Differences in the variants called by the different tools

Because the tools differ in their underlying assumptions about the data, different variants may be called on the same data set using the same filter settings (see section 22.17). In general,

- the **Basic Variant Detection** tool calls the highest number of variants. It runs relatively quickly because it does not do any error-model estimation.
- the **Low Frequency Variant Detection** tool calls only a subset of the variants called by the Basic Variant Detection tool. The variants called by the Basic Variant Caller but **not** called by the Low Frequency Variant Detection tool usually originate from sequencing errors. The Low Frequency Variant caller is the slowest of the three variant callers as it estimates an error-model and does not just consider variants within a specified ploidy model.
- the **Fixed Ploidy Variant Detection** tool calls a subset of the variants called by the Low Frequency Variant Detection tool. The variants called by the Low Frequency Variant Caller but **not** called by the Fixed Ploidy Variant caller likely originate from mapping or sequencing errors.

The following examples show a Track list view of the variants detected by the three different variant detection tools for a particular data set with the same the filter settings. The top three variant tracks contain the results of the variant detection tools. The numbers of variants called are shown on the left side in brackets under the variant track names. The track 'basicV2' contains the results of the Basic Variant Detection tool, the track 'LowFreq' contains the results of the Low Frequencey Variant Detection tool and the track 'FixedV2' contains the results of the Fixed Ploidy Variant detection tool. The other variant tracks display comparisons between results of the different tools. The particular comparisons is described in the name of each of these tracks.

Figure 22.71 highlights a variant reported by the Basic Variant Detection tool but not by the other variant caller tools. The information in the table view of the Basic Variant Detection results track ('basicV2') reveals that the variant is present at a low frequency (3 reads) in a high coverage position (209 reads), suggesting that is not a true variant but rather a sequencing error.

Figure 22.72 shows variant calls produced by the three variant callers with the same data and general filter settings. As expected, the Basic Variant Detection tool reports the most variants (884), the Fixed Ploidy reports the fewest (233), and the Low Frequency Variant Detection tool detects a number between these two (796). But note that in the track named 'inLowFreqV2-notInBasicV2' that there are 9 variants reported by the Low Frequency Variant Detection tool that are not reported by the Basic Variant detection tool. It is because these variants are considered as several SNVs by the Low Frequency Variant Detection tool when they were part of a more complex MNV in the Basic Variant Detection results. In the case of the variant highlighted in figure 22.72, the Low Frequency Variant Detection calls for one variant in results track ('lowFreq'), while the Basic Variant Detection called a heterozygous 2 bp MNV in results track ('basicV2'). Here, the Low Frequency Variant Detection tool called only one of the two SNVs of that MNV. The second SNV of the MNV was not deemed to be supported by the evidence in the data when error modelling was carried out and so was not reported.

Figure 22.73 shows a variant that is detected by both the Basic and the Low Frequency Variant Detection tools, but not by the Fixed Ploidy Variant Detection tool when a ploidy of 2 was

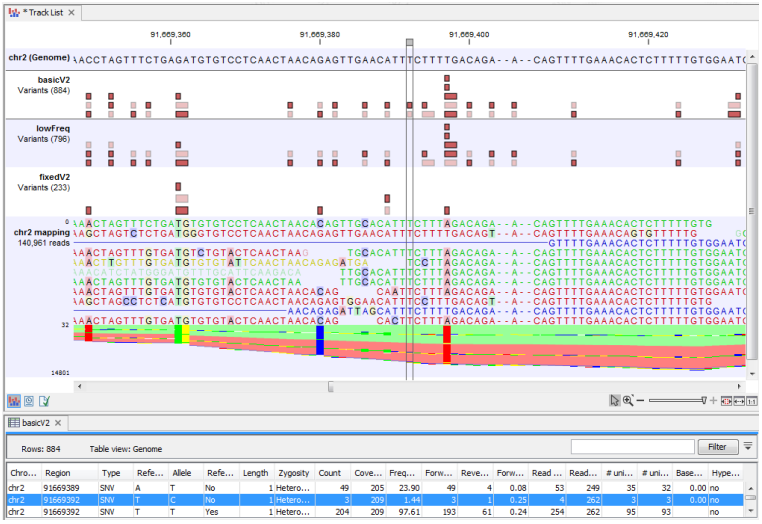


Figure 22.71: *Case where a variant is detected only using the Basic Variant Detection tool.*

Figure 22.72: *Case where variants can be detected as SNV by a tool and MNV by another.*

specified. The information in the table view of the Low Frequency Variant Detection results track ('lowFreq') reveals that the highlighted variant is present in 29 reads in an area with coverage 204, a ratio inconsistent with what can be expected from a diploid sample, thus preventing the stringent Fixed Ploidy Variant Detection tool to call it as a variant. It is also unlikely that this variant was caused by sequencing error. The most likely explanation for the presence of this variant is that it originated from an error in the mapping of the reads. This happens if reads are mapped to a reference area that does not represent their true source, using for example an incomplete reference or one from a too distantly related organism.

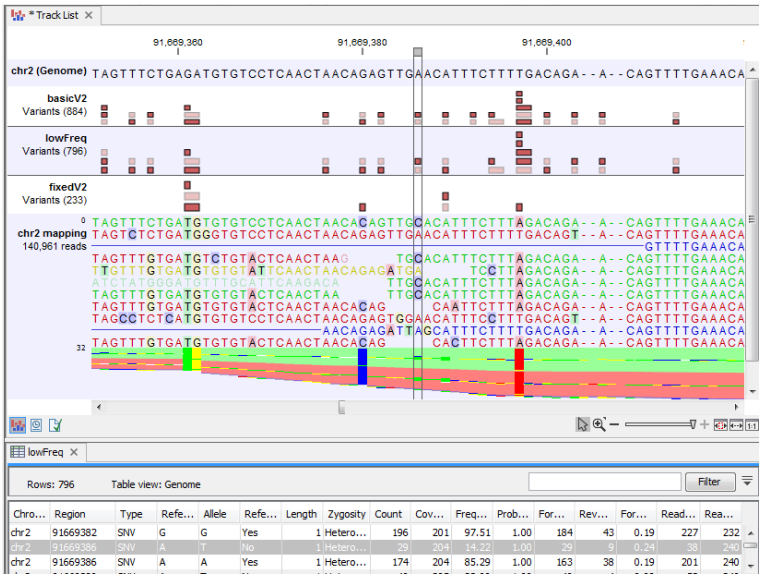


Figure 22.73: *Case where a variant does not fit the ploidy assumption.*

22.12.2 How the variant detection tools work

Each variant detection tool operates in a similar fashion, following successive and iterative steps while using common filters to call for variants. Before you start the tool, the wizard will take you through the different filters you can set to define which of the single polymorphims detected should be called as a variant. The following sections describe the individual characteristics and the specific assumptions of the three variant detection tools. The filtering and output options common to the tools are described in detail in section 22.17 and section 22.18.

The steps of the Variant Detection tools are as follow:

1. The tool identifies all possible variants from either the total input dataset or a subset of it, depending on how the following filters have been set:

   - **Reference masking** settings select the areas of the mapping that should be inspected for variants.
   - **Read filter** settings select for the reads that should be considered in the assessment.
   - **Count and coverage filters** select for sites meeting coverage, frequency and absolute count requirements set for the analysis. Half the value of each parameter is used During the first stage of variant detection, when single position variants are initially being considered. This ensures that multiple position variants, which are built up from the single position variants, are not missed due to too stringent filtering early on. The full values for the cut-offs are applied later during the variant detection process.
   - **Noise filters** specify requirements for a read to be included, considering the quality and neighborhood composition of the area surrounding a potential variant.

2. At this stage, for the Fixed Ploidy and Low Frequency Variant Detection tools only, site-specific information is used to iteratively estimate error models. These error models are then used to distinguish true variants from likely sequencing errors. Potential single nucleotide variants are only be kept if the model containing the variant is significantly better than the model without the variant. Full details for the Fixed Ploidy Variant Detection tool are given in section 22.13 and 22.14.

3. The tool checks each position for other features such as read direction, base qualities and so on using the cut-off values specified in the **Noise filters** (see section 22.17).

4. The tool checks for complex variants by taking the single position variants identified in the steps above and checking if neighboring variants are present in *the same read*. If so, the tool 'joins' these SNVs into MNVs, longer insertions or deletions, or into replacements. Note that SNVs are joined only when they are present in the same read as this provides evidence that the variants appear contiguously in the sample.

5. Finally the tool applies the full cut-off values supplied for the **Count and coverage filters** to the single and multiple position variants obtained during the previous step.

22.13 Fixed Ploidy Variant Detection

The Fixed Ploidy Variant Detection tool relies on two models:

1. A model for the possible 'site-types' depends on the user-specified ploidy parameter: For a diploid organism there are two alleles and thus the site types are A/A, A/C, A/G, A/T, A/-, C/C, and so on until -/-.

2. A model for the sequencing errors that specifies the probabilities of having a certain base in the read but calling a different base. The error model is estimated from the data prior to calling the variants (see section 22.16).

The Fixed Ploidy algorithm will, given the estimated error model and the data observed in the site, calculate the probabilities of each of the site types. One of those site types is the site that is homozygous for the reference - that is, it stipulates that whatever differences are observed from the reference nucleotide in the reads is due to sequencing errors. The remaining site-types are those which stipulate that at least one of the alleles in the sample is different from the reference. The sum of the probabilities for these latter site types is the posterior probability that the sample contains at least one allele that differs from the reference at this site. We refer to this posterior probability as the 'variant probability'.

The Fixed Ploidy Variant Detection tool has two parameters: the 'Ploidy' and the 'Variant probability' parameters (figure 22.74):

- The 'ploidy' is the ploidy of the analyzed sample. The value that the user sets for this parameter determines the site types that are considered in the model. For more information about ploidy please see section 22.13.1.

- The 'Required variant probability' is the minimum probability value of the 'variant site' required for the variant to be called. Note that it is not the minimum value of the probability of the individual variant. For the Fixed Ploidy Variant detector, if a variant site - and not the variant itself - passes the variant probability threshold, then the variant with the highest probability at that site will be reported even if the probability of that particular variant might be less than the threshold. For example if the required variant probability is set to 0.9 then the individual probability of the variant called might be less than 0.9 as long as the probability of the entire variant site is greater than 0.9.

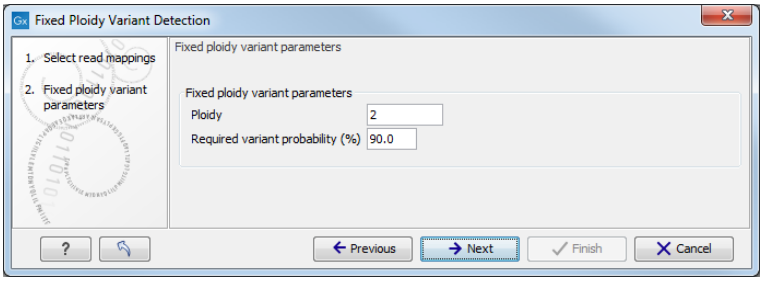


Figure 22.74: *The Fixed Ploidy Variant Detection parameters.*

As the Fixed Ploidy Variant Detection tool strongly depends on the model assumed for the ploidy, the user should carefully consider the validity of the ploidy assumption that he makes for his sample. The tool allows ploidy values up to and including 4 (tetraploids). For higher ploidy values the number of possible site types is too large for estimation and computation to be feasible, and the user should use the Low Frequency or Basic Variant Detection Tool instead.

For a more in depth description of the Fixed Ploidy variant caller see Section 22.19.

22.13.1 Ploidy and sensitivity

The Fixed Ploidy Variant Detection tool has two parameters. The ploidy level you set defines the statistical model that will be used during the variant detection analysis and thereby also defines what will be reported. The number of alleles that variant may have depends on the value that has been chosen for the ploidy parameter. For example, if you chose a ploidy of 2, then the variant at a site could be a homozygote (two alleles the same in the sample, but different to the reference), or a heterozygote (two alleles different than each other in the sample, with at least one of them different from the reference). If you had chosen a ploidy of three, then the variant at a site could be a homozygote (three alleles the same in the sample, but different to the reference), or a heterozygote (three alleles different than each other in the sample, with at least one of them different from the reference).

The variant probability parameter defines how good the evidence has to be at a particular site for the tool to report a variant at that location. If the *site* passes this threshold, then the *variant* with the highest probability at that site will be reported.

Sensitivity of the tool can be altered by changing these parameters: to increase sensitivity, you could decrease the variant probability setting - more sites are being reported - or increase the ploidy - adding extra allele types.

For example, a sample with a ploidy of 2 has many C and a few G at a particular location where the reference is a T. There is high enough evidence that the actual position is different than the reference, so the variant with the highest probability at this location will be reported. In the diploid model, all the possibilities will have been tested (e.g. A|A, A|C....C|C, C|G. C|T....and so on). In this example, C|C had the highest probability, and as long as the relative prevalence of Gs is low compared to Cs - that is, the probability of C|C stays higher than C|G - C|C will be reported. But in a case where the sample has a ploidy of 3, the model will test all the triploid possibilities (e.g. A|A|A, A|A|C, A|A|G.....C|C|A, C|C|C, C|C|G.... and so on). For the same site, if the evidence in the reads results in the variant C|C|G having a higher probability than C|C|C, then it would be the variant reported. This shows that by increasing ploidy we have increased sensitivity of the tool, reporting a variant that represents the reads with G as well as the ones reporting a C at a particular position.

22.14 Low Frequency Variant Detection

As the Fixed Ploidy Variant Detection tool, the Low Frequency Variant Detection tool relies on

1. A statistical model for the analyzed sample and
2. A model for the sequencing errors.

A statistical test is performed at each site to determine if the nucleotides observed in the reads at that site could be due simply to sequencing errors, or if they are significantly better explained by there being one (or more) alleles. If the latter is the case, a variant corresponding to the significant allele will be called with an estimated frequency.

The Low Frequency Variant Detection tool has one parameter (figure 22.75):

- **Required Significance**: this parameter determines the cut-off value for the statistical test for the variant not being due to sequencing errors. Only variants that are at least this significant will be called. The lower you set this cut-off, the fewer variants will be called.

The Low Frequency Variant Detection tool is suitable for analysis of samples of mixed tissue types (such as cancer samples) in which low frequent variants are likely to be present, as well as for samples for which the ploidy is unknown or not well defined. The tool also calls more abundant variants, and can be used for analysis of samples with ploidy larger than four. Note that, as the tool looks for all variants, abundant as well as low frequency ones, analysis will generally be slower than those of the other variant detection tools. In particular it will be very slow - possibly prohibitively so - for samples with extremely high coverage, or a very large number of variants (as in cases where the sample differs considerably from the reference).

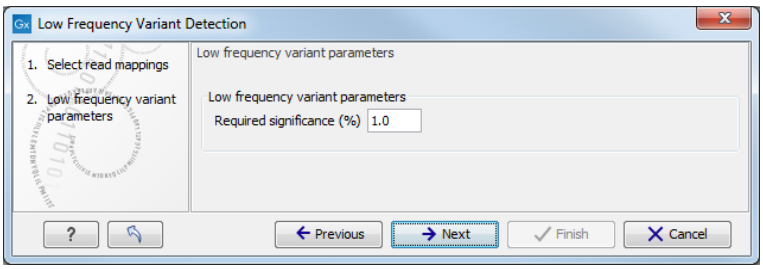


Figure 22.75: *The Low Frequency Variant Detection parameters.*

For a more in depth description of the Low Frequency variant caller see section 22.19.

22.15 Basic Variant Detection

The Basic Variant Detection tool does not rely on any assumptions on the data, and does not estimate any error model. It can be used on any type of sample. It will call a variant if it satisfies the requirements that you specify when you set the filters (see section 22.17). The tool has a single parameter (figure 22.76) that is specific to this tool: the user is asked to specify the 'ploidy' of the sample that is being analyzed. The value of this parameter does not have an impact on which variants are called - it will merely determine the contents of the 'hyper-allelic' column that is added to the variant track table: variants that occur in positions with more variants than expected given the specified ploidy, will have 'Yes' in this column, other variants will have 'No' (see section 22.18 for a description of the outputs).

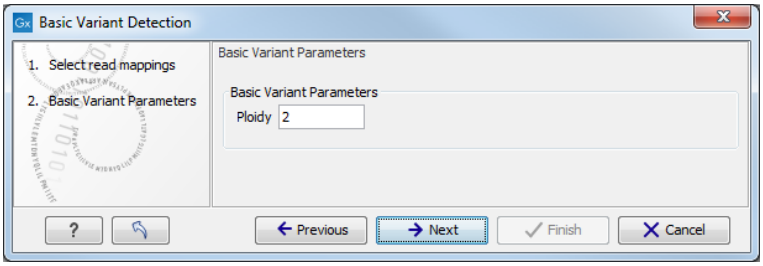


Figure 22.76: *The Basic Variant Detection parameters.*

22.16 Variant Detectors - error model estimation

The Fixed Ploidy and Low Frequency Variant Detection tools both rely on statistical models for the sequencing error rates. An error model is assumed and estimated for each quality score. Typically low quality read nucleotides will have a higher error rate than high quality nucleotides. In the error models, different types of errors have their own parameter, so if A's for example more often tend to result in erroneous G's than other nucleotides, that is also recognized by the error models. The parameters are all estimated from the data set being analyzed, so will adapt to the sequencing technology used and the characteristics of the particular sequencing runs. Information on the estimated error rates can be found in the Reports (see section 22.18 and figure 22.77).

1.5 Estimated frequencies of actual to called bases (quality scores: 20-29)

Called (across): Actual (below):	A	C	G	T	N	-
A	99.828	0.015	0.086	0.023	0.044	0.005
C	0.050	99.854	0.034	0.043	0.017	0.002
G	0.043	0.011	99.897	0.029	0.017	0.003
T	0.026	0.050	0.032	99.868	0.021	0.003
-	0.000	0.000	0.000	0.000	0.000	100.000

Number of sequenced bases with quality scores 20-29: 382,854,867

1.6 Estimated frequencies of actual to called bases (quality scores: 30-39)

Called (across): Actual (below):	A	C	G	T	N	-
A	99.979	0.001	0.008	0.003	0.008	0.001
C	0.010	99.976	0.002	0.008	0.002	0.001
G	0.008	0.001	99.974	0.012	0.004	0.001
T	0.003	0.008	0.002	99.983	0.003	0.001
-	0.000	0.000	0.000	0.000	0.000	100.000

Number of sequenced bases with quality scores 30-39: 7,400,088,878

Figure 22.77: *Example of estimated error rates estimated from a whole exome sequencing Illumina data set.*

The figure shows average estimated error rates across bases in the given quality score intervals (20-29 and 30-39, respectively).As expected, the estimated error rates (that is, the off-diagonal elements in the matrices in the figure) are higher for bases with lower quality scores. Note also that although the matrices in the figure show error rates of bases within *ranges of* quality scores, a separate matrix is estimated for each quality score in the error model estimation.

22.17 Variant Detectors - filters

The variant detectors offer a number of filters for which the user will set values in two wizard steps, the 'General filters' and the 'Noise filters'. Note that the tools add for most filters the values calculated and the bases filtered on as annotations to the variants (see section 22.18). This means that the filtering information is available in the variant track and that the user can

choose to perform the filtering in a post-processing step from the variant track table rather than applying the filtering during the variant calling detection step.

The filters are described below.

22.17.1 General filters

The General filters relate to the regions and reads in the read mappings that should be considered, and the amount of evidence the user wants to require for a variant to be called (figure 22.78):

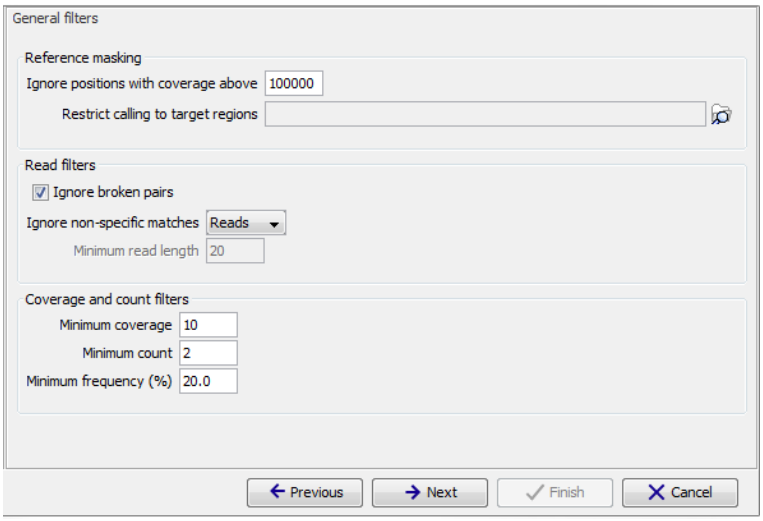


Figure 22.78: *General filters. The values shown are those that are default for Fixed Ploidy Variant detection.*

Note on the use of the Low Frequency Variant Detection tool with Whole Genome Sequencing data: The default settings for the Low Frequency Variant caller are optimized for targeted resequencing protocols, and NOT whole genome sequencing (e.g. cancer gene panels) where it is not uncommon to have modest coverage for most part of the mapping, and abnormal areas (typically repeats around the centromeres) with very high coverage. Looking for low frequency variants in high coverage areas will exhaust the machine memory because there will be many low frequency variants due to some reads originating from near identical repeat sequences or simple sequencing errors.In order to run the tool on WGS data the parameter **Ignore positions with coverage above** should be adjusted to a lower number (typically 1000).

Reference masking

The 'Reference masking' filters allow the user to only perform variant calling (including error model estimation) in specific regions. In addition to selecting an annotation track, there are two parameters to specify:

- **Ignore positions with coverage above:** All positions with coverage above this value will be ignored when inspecting the read mapping for variants. The option is highly useful in cases where you have a read mapping which has areas of extremely high coverage as are areas around centromeres in whole genome sequencing applications for example.
- **Restrict calling to target regions:** Only positions in the regions specified will be inspected for variants.

Read filters

The Read filters determine which reads (or regions) should be considered when calling the variants.

- **Ignore broken pairs:** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected. Please note that ignored broken pair reads will not be considered for any non-specific match filters.
- **Non-specific match filter:** Non-specific matches are likely to come from repeat region whose exact mapping location is uncertain. In general, variants based on non-specific matches are likely to be less reliable. However, as there are regions in the genome that are entirely perfect repeats, ignoring non-specific matches may have the effect that true variants go undetected in these regions.

  There are three options for specifying to which 'extent' the non-specific matches should be ignored:

  - 'No': they are not ignored.
  - 'Reads': they are ignored.
  - 'Region': when this option is chosen no variants are called in regions covered by at least one non-specific match. In this case, the minimum length of reads that are allowed to trigger this effect has to be stated, as really short reads will usually be non-specific even if they do not stem from repeat regions.

Coverage and count filters

These filters specify absolute requirements for the variants to be called. Note that suitable values for these filters are highly dependent on the coverage in the sample being analyzed:

- **Minimum coverage:** Only variants in regions covered by at least this many reads are called.
- **Minimum count:** Only variants that are present in at least this many reads are called.
- **Minimum frequency:** Only variants that are present at least at the specified frequency (calculated as 'count'/'coverage') are called.

These values are calculated for each of the detected candidate variants. If the candidate variant meets the specified requirements, it is called. Note that when the values are calculated, only the 'countable reads' - the reads chosen by the user to NOT be ignored - are considered. For example, if the user had specified to ignore reads from broken pairs, they will not be countable. This is also the case for non-specific reads, and for reads with bases at the variant position that does not fulfill the base quality requirements specified by the 'Base Quality Filter' (see the section on 'Noise filters' below). Also note that overlapping paired reads only count as one read since they only represent one fragment.

22.17.2 Noise filters

The 'Noise filters' examine each candidate variant at a more detailed level and filter out those that are likely the result of systematic errors and/or biases, such as those coming from samples preparation steps or sequencing protocol (figure 22.79). These filters should be used with care as there is always the risk of not calling a real variant that has the characteristics of a systematically induced one.

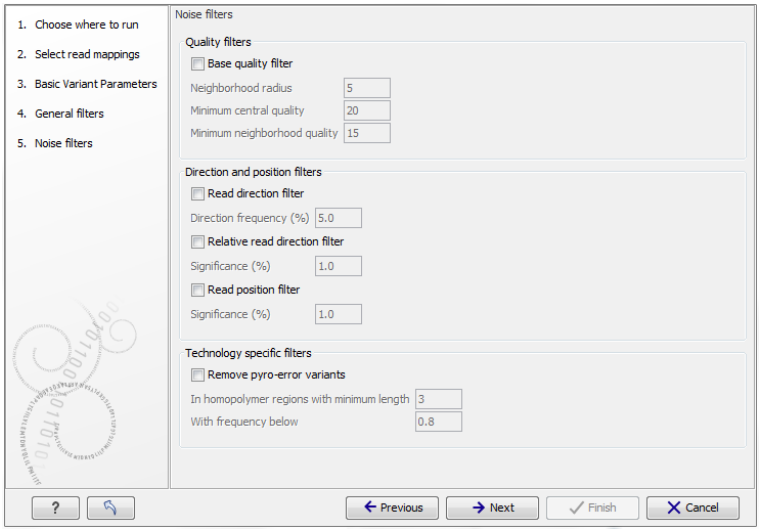


Figure 22.79: *Noise filters.*

Quality filters

- **Base quality filter:** The base quality filter can be used to ignore the reads whose nucleotide at the potential variant position is of dubious quality. This is assessed by considering the quality of the nucleotides in the region around the nucleotide position. There are three parameters to determine the base quality filter:
  - **Neighborhood radius**: This parameter determines the region size. For example if a neighborhood radius of five is used, a nucleotide will be evaluated based on the nucleotides that are 5 positions upstream and 5 positions downstream of the examined site, for a total of 11 nucleotides. Note that, near the end of the reads, eleven nucleotides will still be considered by offsetting the region relative to the nucleotide in question.
  - **Minimum central quality:** Reads whose central base has a quality below the specified value will be ignored. This parameter does not apply to deletions since there is no 'central base' in these cases.
  - **Minimum neighborhood quality:** Reads for which the minimum quality of the bases is below the specified value will be ignored.

Figure 22.80 gives an example of a variant called when the base quality filter is NOT applied, and not called when it is. When switching on the 'Show quality scores' option in the side panel of the reads it becomes visible that the reads that carry the potential 'G' variant tend to have poor quality. Note that the error in the example shown is a 'typical' Illumina error: the reference has a 'T' that is surrounded by stretches of 'G', the 'G' signals 'drowning' the signal of the 'T'. As

all reads that have a base with quality less than 20 in this potential variant position are ignored when the 'Base quality filter' is turned on, no variant is called, most likely because it now does not meet the requirements of either the 'Minimum coverage', 'Minimum count' or 'Minimum frequency' filters.

Figure 22.80: *Example of a variant called when the base quality filter is NOT applied, and not called when it is.*

Direction and position filters

Many sequencing protocols are prone to various types of amplification induced biases and errors. The 'Read direction' and 'Read position' filters are aimed at providing means for weeding out variants that are likely to originate from such biases.

- **Read direction filter:** The read direction filter removes variants that are almost exclusively present in either forward or reverse reads. For many sequencing protocols such variants are most likely to be the result of amplification induced errors. Note, however, that the filter is **NOT suitable for amplicon data**, as for this you will not expect coverage of both forward and reverse reads. The filter has a single parameter:
  - **Direction frequency:** Variants that are not supported by at least this frequency of reads from each direction are removed.
- **Relative read direction filter:** The relative read direction filter attempts to do the same thing as the 'Read direction filter', but does this in a statistical, rather than absolute, sense: it tests whether the distribution among forward and reverse reads of the variant carrying reads is different from that of the total set of reads covering the site. The statistical, rather than absolute, approach makes the filter less stringent. The filter has one parameter:
  - **Significance:** Variants whose read direction distribution is significantly different from the expected with a test at this level are removed. The lower you set the significance cut-off, the fewer variants will be filtered out.
- **Read position filter:** The read position filter is a filter that attempts to remove systematic errors in a similar fashion as the 'Read direction filter', *but* that is also **suitable for hybridization-based data**. It removes variants that are located differently in the reads carrying it than would be expected given the general location of the reads covering the variant site. This is done by categorizing each sequenced nucleotide (or gap) according

to the mapping direction of the read and also where in the read the nucleotide is found; each read is divided in five parts along its length and the part number of the nucleotide is recorded. This gives a total of ten categories for each sequenced nucleotide and a given site will have a distribution between these ten categories for the reads covering the site. If a variant is present in the site, you would expect the variant nucleotides to follow the same distribution. The read position filter carries out a test for whether the read position distribution of the variant carrying reads is different from that of the total set of reads covering the site. The filter has one parameter:

- **Significance:** Variants whose read position distribution is significantly different from the expected with a test at this level, are removed. The lower you set the significance cut-off, the fewer variants will be filtered out.

Figure 22.81 shows an example of a variant that is removed by the 'Read direction' filter. To see the direction of the reads, you must adjust the viewer settings in the 'Reads track' side panel to 'Disconnect paired reads'. Note that variant calling was done ignoring non-specific matches and broken pair reads, so only the 16 intact forward paired reads (the green reads) are considered. In this example there was no intact reverse reads.

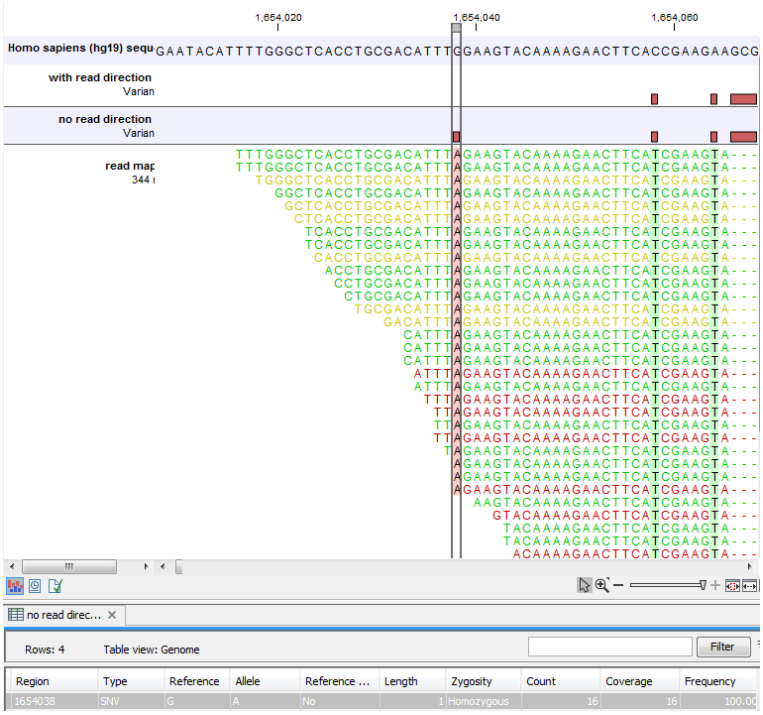


Figure 22.81: *Example of a variant that is removed by the 'Read direction' filter.*

Figure 22.82 shows an example of a variant that is removed by the 'Read position' filter, but not by the 'Read direction' filter. This variant is only seen in a set of reads having a similar start position, while reads that start in a different location do not contain this variant (e.g., none of the reads that start after position 186,641,600 carry the variant). This could indicate the incorporation of an incorrect base during the library preparation process rather than a true biological variant. The purpose of the 'Read position' filter is to reduce the presence of these types of variants. As with all noise filters, the more stringent the setting, the more likely you are to remove false positives and enrich your result for true positive variant calls but comes with the risk of filtering out true positives as well.

Understanding the type of false positive this filter is intended to remove will help you to determine what makes sense for your data set. For example, if your sequencing data did not include a PCR step or hybrid capture step, you may wish to use more lax settings for this filter (or not use it at all).

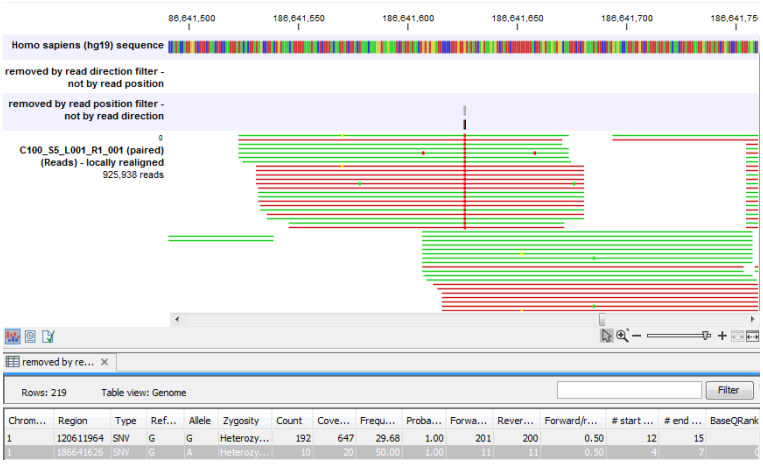


Figure 22.82: *A variant that is filtered out by the Read position filter but not by the Read direction filter.*

Technology specific filters

- **Remove pyro-error variants:** This filter can be used to remove insertions and deletions in the reads that are likely to be due to pyro-like errors in homopolymer regions. There are two types of such errors: They may occur either at (1) the immediate ends of homopolymer regions or (2) as an 'overspill' a few nucleotides downstream of a homopolymer region. In case (1) the exact numbers of the same number of nucleotide is uncertain and a sequence like "AAAAAAAA" is sometimes reported as "AAAAAAAA**A**". In case (2) a sequence like "CGAAAAAGTCG" may sometimes get an 'overspill' insertion of an A between the T and C so that the reported sequence is C "CGAAAAAGT**A**CG". Note that the removal is done in the reads as a very first step, before calling the initial 1 bp variants.

  There are two parameters that must be specified for this filter:

  - **In homopolymer regions with minimum length**: Only insertion or deletion variants in homopolymer regions of at least this length will be removed.
  - **With frequency below:** Only insertion or deletion variants whose frequency (ignoring all non-reference and non-homopolymer variant reads) is lower than this threshold will be removed.

Note that the higher you set the **With frequency below** parameter, the more variants will be removed. Figure 22.83 shows an example of a variant that is called when the pyro-error filter with minimum length setting 3 and frequency setting 0.5 is used, but that is filtered when the frequency setting is increased to 0.8. The variant has a frequency of 55.71.

In addition to the example above, a simple example is provided below in figure 22.84 to illustrate the difference between variant frequency and pyro-variant removal frequency (where non-reference and non-homopolymer variant reads are ignored).

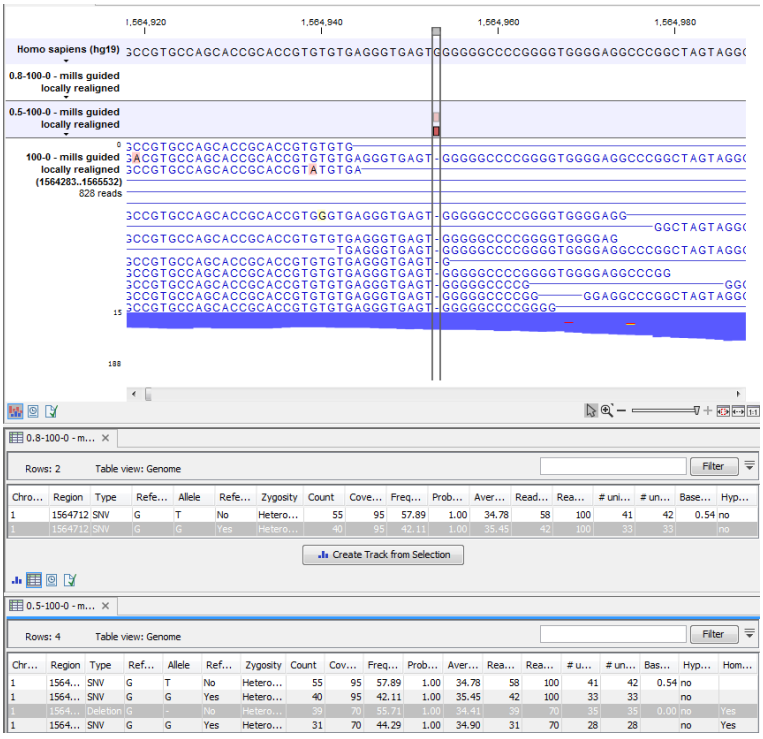


Figure 22.83: *An example of a variant that is filtered out when the pyro-error filter is applied with settings 3 and 0.8, but not with settings 3 and 0.5.*

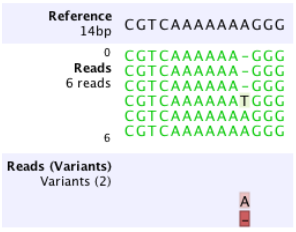


Figure 22.84: *An example of a simple read mapping with 6 mapped reads. Three of them indicate a deletion, two match the reference, and one read is an A to T SNP*

The read with the T variant is not counted when calculating the frequency for the homopolymer deletion, because we only want to estimate how often a homopolymer variant appears for a given allele, and the T read is not from the same allele as the A and gap reads.

For the deletion, the variant frequency will be 50 percent, if it is reported. This is because it appears in 3 of 6 reads.

However, the pyro-variant removal frequency is 0.6, because it appears in 3 of 5 reads that come from the same allele. Thus the deletion will only be removed by the pyro-filter if the **With frequency below** parameter is above 0.6 and the **In homopolymer regions with minimum length** parameter is less than 7.

22.18 Variant Detectors - the outputs

The Variant Detection Tools have the following outputs: a variant track, an annotated variant table and a report (figure 22.85). The report contains information on the estimated error model and, as only the Fixed ploidy and the Low Frequency variant callers uses an error model, the report is only available for those, and not for the Basic Variant caller. The outputs are described below.

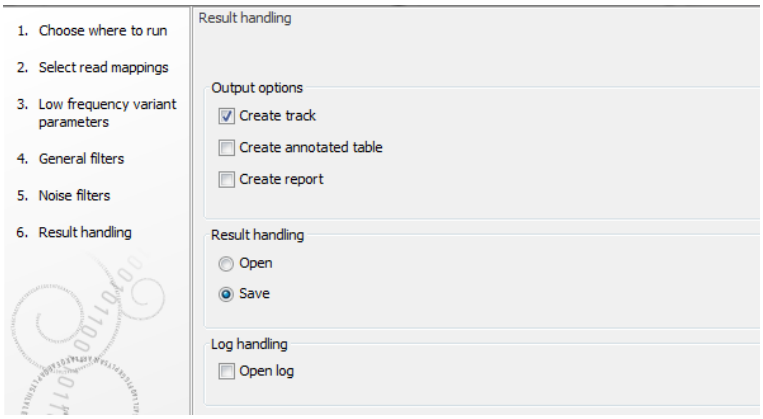


Figure 22.85: *Output options.*

22.18.1 The variant track output

The variant track contains information on each of the variants called, including reference alleles. When opened in the table view there is a number of columns for each of the variants (see figure 22.86).

Rows: 74,865 Table view: Genome Filter

Ch...	Region	Type	Refe...	Allele	Refer...	Length	Zygosity	Count	Coverage	Freque...	Proba...	Forwa...	Reve...	Forward/r...	Avera...	Read c...	Read cov...	# uniq...	# uniq...	Base...	Hy...	...
1	1564953	Dele...	G	-	No	1	Heterozyg...	39	70	55.71	1.00	34	5	0.13	34.41	39	70	35	35	0.00	no	Yes
1	1564953	SNV	G	G	Yes	1	Heterozyg...	31	70	44.29	1.00	24	7	0.23	34.90	31	70	28	28		no	Yes
1	1920434^1...	Inse...	-	A	No	1	Heterozyg...	5	24	20.83	0.90	2	4	0.33	32.00	6	27	5	5	0.00	no	Yes
1	1920434^1...	Inse...	-	-	Yes	0	Heterozyg...	18	24	75.00	0.90	13	6	0.32	35.06	20	27	17	16		no	Yes
1	3395974..3...	MNV	CC	CC	Yes	2	Heterozyg...	23	42	54.76	1.00	20	6	0.23	35.96	26	47	36	39		no	Yes
1	3395974..3...	Dele...	CC	-	No	2	Heterozyg...	19	42	45.24	1.00	13	8	0.38	31.58	21	47	36	39	2.86	no	Yes

Figure 22.86: *A variant track shown in the table view.*

The contents of these are:

Chromosome The name of the reference sequence on which the variant is located.

Region The region on the reference sequence at which the variant is located. The region may be either a 'single position', a 'region' or a 'between position region'.

Type The type of variant. This can either be SNV (single-nucleotide variant), MNV (multi-nucleotide variant), insertion, deletion, or replacement. Learn more in section 22.20.3.

Reference The reference sequence at the position of the variant.

Allele The allele sequence of the variant.

Reference allele Describes whether the variant is identical to the reference. This will be the case for one of the alleles for most, but not all, detected heterozygous variants (e.g. the variant caller might detect two variants, A and G, at a given position in which the reference is 'A'. In this case the variant corresponding to allele 'A' will have 'Yes' in the 'reference

allele' column entry, and the variant corresponding to allele 'G' would have 'No'. Had the variant caller called the two variants 'C' and 'G' at the position, both would have had 'No' in the 'Reference allele' column).

Length The length of the variant. The length is 1 for SNVs, and for MNVs it is the number of allele or reference bases (which will always be the same). For deletions, it is the length of the deleted sequence, and for insertions it is the length of the inserted sequence. For replacements, both the length of the replaced reference sequence and the length of the inserted sequence are considered, and the longest of those two is reported.

Zygosity The zygosity of the variant called, as determined by the variant caller. This will be either 'Homozygous', where there is only one variant called at that position or 'Heterozygous' where more than one variant was called at that position.

Count The number of 'countable' fragments supporting the allele. The 'countable' fragments are those that are used by the variant caller when calling the variant. Which fragments are 'countable' depends on the user settings when the variant calling is performed - for example, if the user has chosen 'Ignore broken pairs', reads belonging to broken pairs are not 'countable'. Note that, although overlapping paired reads have two reads in their overlap region, they only represent one fragment, and are counted only as one. (Please see the column 'Read count' below for a column that reports the value for 'reads' rather than for 'fragments'). Note also that the count value reported in the table may differ from the one accessible from the track's tooltip, as the 'count' value in the table is generated taking into account quality score and frequency of sequencing errors.

Coverage The fragment coverage at this position. Only 'countable' fragments are considered (see under 'Count' above for an explanation of 'countable' fragments). Note that, although overlapping paired reads have two reads in their overlap region, they only represent one fragment, and overlapping paired reads contribute only 1 to the coverage. (Please see the column 'Read coverage' below for a column that reports the value for 'reads' rather than for 'fragments'). Also see overlapping pairs in section 22.21 for how overlapping paired reads are treated.)

Frequency 'Count' divided by 'Coverage'.

Probability The contents of the Probability column (for Low Frequency and Fixed Ploidy variant callers only) depend on the variant caller that produced and the type of variant:

- In the Fixed Ploidy Variant Detection Tool, the probability in the resulting variant track's 'Probability' column is NOT the probability referred to in the wizard. The probability referred to in the wizard is the required minimum (posterior) probability that the site is NOT homozygous for the reference. The probability in the variant track 'Probability' column is the posterior probability of the particular site-type called. The fixed ploidy tool calculates the probability of the different possible configurations at each site. So using this tool, for single site variants the probability column just contains this quantity (for variants that span multiple positions see below).
- The Low Frequency Variant Detection tool makes statistical tests for the various possible explanations for each site. This means that the probability for the called variant must be estimated separately since it is not part of the actual variant calling. This is done by assigning prior probabilities to the various explanations for a site in a way that makes the probability for two explanations equal in exactly the situation

where the statistical test shifts from preferring one explanation to the other. For a given single site variant, the probability is then calculated as the sum of probabilities for all the explanations containing that variant. So if a G variant is called, the reported probability is the sum of probabilities for these configurations: G, A/G, C/G, G/T, A/C/G, A/G/T, C/G/T, and A/C/G/T (and also all the configurations containing deletions together with G).

For multi position variants, an estimate is made of the probability of observing the same read data if the variant did not exist and all observations of the variant were due to sequencing errors. This is possible since a sequencing error model is found for both the fixed ploidy and rare variant tools. The probability column contains one minus this estimated probability. If this value is less than 50%, the variant might as well just be the result of sequencing errors and it is not reported at all.

Forward read count The number of 'countable' forward reads supporting the allele (see under 'Count' above for an explanation of 'countable' reads).

Reverse read count The number of 'countable' reverse reads supporting the allele (see under 'Count' above for an explanation of 'countable' reads).

Forward/reverse balance The minimum of the fraction of 'countable' forward reads and 'countable' reverse reads carrying the variant among all 'countable' reads carrying the variant (see under 'Count' above for an explanation of 'countable' reads).[1]

Average quality The average base quality score of the bases supporting a variant. In the case of a deletion, the quality score is taken from the average quality of the two bases neighboring the deleted one, and the lowest is reported. Similarly for insertions, the quality in reads where the insertion is absent is taken from the minimum average of the two bases on either side of the position. It can be possible in rare cases, that the quality score reported in this column for a deletion or insertion is below the threshold set for 'Minimum central quality', because this parameter is not applied to any quality value calculated from positions *outside* of the central variant. If there are no values in this column, it is probably because the sequencing data was imported without quality scores (learn more about importing quality scores from different sequencing platforms in section 6.3).

Read count The number of 'countable' reads supporting the allele. Only 'countable' reads are considered (see under 'Count' above for an explanation of 'countable' reads). Note that each read in an overlapping pair contribute 1. To view the reads in pairs in a reads track as single reads, check the 'Disconnect paired reads' option in the side-panel of the reads track. (Please see the column 'Count' above for a column that reports the value for 'fragments' rather than for 'reads').

Read coverage The read coverage at this position. Only 'countable' reads are considered (see under 'Count' above for an explanation of 'countable' reads). Note that each read in an overlapping pair contribute 1. To view the reads in pairs in a reads track as single reads, check the 'Disconnect paired reads' option in the side-panel of the reads track. (Please

[1] Some systematic sequencing errors can be triggered by a certain combination of bases. This means that sequencing one strand may lead to sequencing errors that are not seen when sequencing the other strand (see [Nguyen et al., 2011] for a recent study with Illumina data). In order to evaluate whether the distribution of forward and reverse reads is approximately random, this value is calculated as the minimum of the number of forward reads divided by the total number of reads and the number of reverse reads divided by the total number of reads supporting the variant. An equal distribution of forward and reverse reads for a given allele would give a value of 0.5.)

see the column 'Coverage' above for a column that reports the value for 'fragments' rather than for 'reads').

 # Unique start positions The number of unique start positions for 'countable' fragments that support the variant. This value can be important to look at in cases with low coverage. If all reads supporting the variant have the same start position, you could suspect that it is a result of an amplification error.

 # Unique end positions The number of unique end positions for 'countable' fragments that support the variant. This value can be important to look at in cases with low coverage. If all reads supporting the variant have the same end position, you could suspect that it is a result of an amplification error.

BaseQRankSum The BaseQRankSum column contains an evaluation of the quality scores in the reads that have a called variant compared with the quality scores of the reference allele. Reference alleles and variants for which no corresponding reference allele is called do not have a BaseQRankSum value. The score is a z-score derived using the Mann-Whitney U test, so a value of -2.0 indicates that the observed qualities for the variant are two standard deviations below what would be expected if they were drawn from the same distribution as the reference allele qualities. A negative BaseQRankSum indicates a variant with lower quality than the reference variant, and a positive z-score indicates higher quality than the reference.

Read position test probability The test probability for the test of whether the distribution of the read positions variant in the variant carrying reads is different from that of all the reads covering the variant position.

Read direction test probability Tests whether the distribution among forward and reverse reads of the variant carrying reads is different from that of all the reads covering the variant position. This value reflects a balanced presence of the variant in forward and reverse reads (1: well-balanced, 0: un-balanced). This p-value is based on a statistic that we assume follows a Chi-square(df=2) distribution under the null hypothesis of the variant having equal frequency on reads from both direction. Note that GATK uses a Fisher's exact test for the same purpose. The difference between both approaches lead to a potential overestimation of p-values output by the workbench's variant callers.

Hyper-allelic Basic and Fixed Ploidy Variant detectors only: Contains "yes", if the site contains more variants than the user-specified ploidy predicts, "no" if not.

Genotype Fixed Ploidy only: Contains the most probable genotype for the site.

Homopolymer The column contains "Yes" if the variant is likely to be a homopolymer error and "No" if not. This is assessed by inspecting all variants in homopolymeric regions longer than 2. A variant will get the mark "yes" if it is a homopolymeric length variation of the reference allele, or a length variation of another variant that is a homopolymeric variation of the reference allele. When several overlapping homopolymeric variants are identified, all except the most frequent are marked as being homopolymer. However, if one of the overlapping, homopolymeric variants is the reference allele, then all of them are marked as homopolymer.

QUAL This value is necessary for certain downstream analyses of the data after export in vcf format. It is calculated as

$$-10\log_{10}(1-p) \tag{22.10}$$

p being the probability that a particular variant exists in the sample (see above for the definition of probability). A QUAL value of 10 indicates a 1 in 10 chance that the called variant is an error, while a QUAL of 100 indicates a 1 in 10^{10} chance that the called variant is an error. QUAL is capped at 200 for p=1.

22.18.2 The annotated table output

The 'Annotated table' contains only non-reference alleles. For a detailed description of the output, please see section 22.20.2.

22.18.3 The report

In addition to the estimated error rates of the different types of errors shown in figure 22.77, the report contains information on the total error rates for each quality score as well as a distribution of the qualities of the individual bases in the reads in the read mapping, at the sites that were examined for variants (see figure 22.87).

22.19 The Fixed Ploidy and Low Frequency variant callers: detailed descriptions

This section provides a detailed description of the models, methods and estimation procedures behind the Fixed Ploidy and Low frequency variant callers. For less detailed descriptions please see sections Section 22.13 and Section 22.14.

22.19.1 The Fixed Ploidy Variant Caller: Models and methods

This section describes the model, method and estimation procedure behind the Fixed Ploidy Variant Caller. The Fixed Ploidy Variant Caller is designed for detecting variants in samples for which the ploidy is known. As the Fixed Ploidy Variant Caller assumes, and hence can exploit, information about underlying possible allele type sites, this variant caller has particularly high *specificity* for samples for which the ploidy assumption is valid.

The Fixed Ploidy Variant Caller

The purpose of the Fixed Ploidy Variant Caller is to call variants in samples with known ploidy from read mapping data. It can detect variants in samples from haploid (e.g. bacteria), diploid (e.g. human) and polyploid (upto tetraploid) organisms (e.g. higher plants). It detects Single Nucleotide Variants (SNVs), MNVs (Multiple Nucleotide Variants), insertions, deletions as well as replacements (combinations of neighboring insertions, deletions and SNVs for which the positions are ambiguous).

The algorithm behind the Fixed Ploidy Variant Caller combines a Bayesian model with a Maximum Likelihood approach. Variants are called by examining the posterior probabilities from the Bayesian model: at any given site a variant is called if the sum of the posterior probabilities of the site types that are different from the homozygous reference allele site type is larger than the user-specified 'probability' cut-off value. The variant called is the variant that corresponds to the site type with the highest posterior probability. When evaluating the posterior probabilities in the

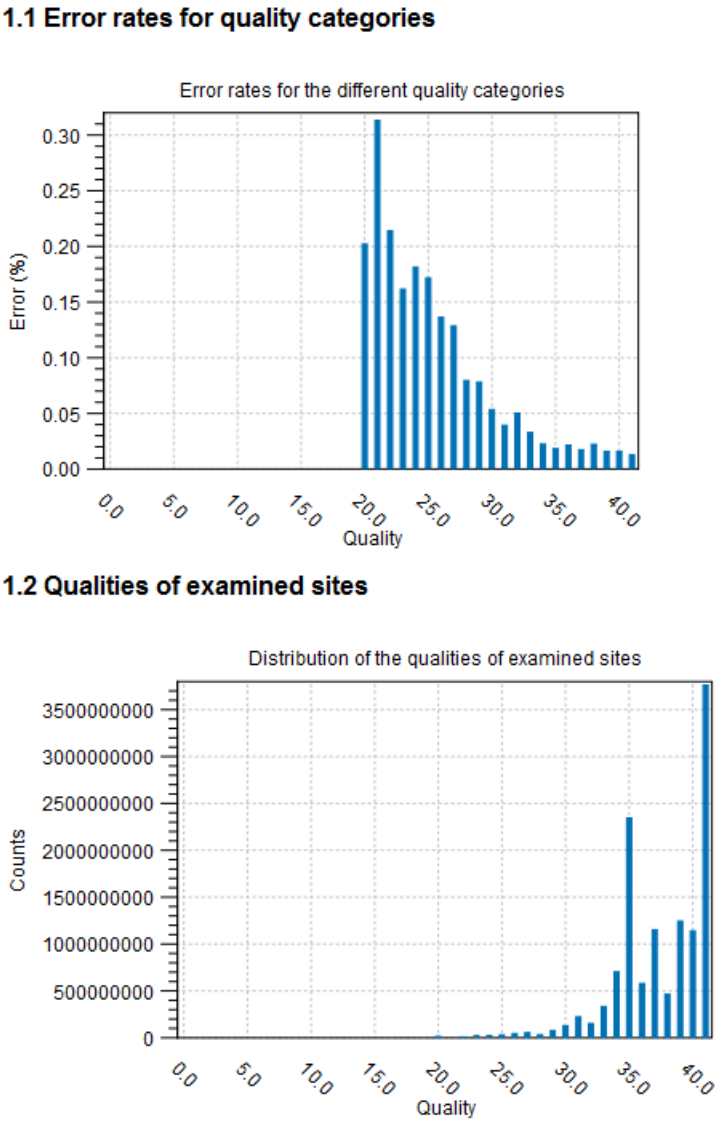


Figure 22.87: *Part of the contents of the report on the variant calling.*

Bayesian model, maximum likelihood estimates for the parameters of the model are used. These are obtained from the likelihood function using an Expectation Maximization (EM) approach.

The model for the Fixed Ploidy Variant Caller

The statistical model for the Fixed Ploidy Variant Caller consists of a model for the **the possible site types**, S, and their prior probabilities, $f_s, s \in S$, and for **the sequencing errors**, e.

Prior site type probabilities: The set of possible site types is determined entirely by the assumed ploidy, and consists of the set of possible underlying nucleotide allele combinations that can exist within an individual with the specified number of alleles. E.g. if the specified ploidy is 2, the individual has two alleles, and the nucleotide at each allele can either be an A, a C, a G, a T or a $-$. The set of possible types for the diploid individual's sites is thus:

$$S = \{A/A, A/C, A/G, A/T, A/-, C/C, C/G, C/T, C/-, G/G, G/T, G/-, T/T, T/-, -/-\}.$$

Note that, as we cannot distinguish the alleles from each other there are not 5 $\times$ 5 = 25 possible site types, but only 15 (that is, the allele combination A/C is indistinguishable from the allele combination C/A).

We let f_s denote the prior probabilities of the site types $s \in S$. The prior probabilities of the site types are the frequencies of the true site types in the mapping. The values of these are unknown, and need to be estimated from the data.

Error probabilities: The model for the sequencing errors describes the probabilities with which the sequencing machine produces the nucleotide M, when what it should have produced was the nucleotide N, (M and $N \in \{A, C, G, T, -\}$). When quality values are available for the nucleotides in the reads, we let each quality value have its own error model; if not, a single model is assumed for all nucleotides. Each error model has the following 25 parameters:

$$\{e(N \to M)|N, M \in \{A, C, G, T, -\}\}.$$

The values of these parameters are also unknown, and hence also need to be estimated from the data.

Deriving the posterior probabilities of the site types

We will call a variant at a site if the sum of the posterior probabilities of the non-homozygous reference site types is larger than the user-specified cut-off value. For this we need to be able to calculate the posterior site type probabilities. We here derive the formula for these.

Using the Bayesian approach we can write the posterior probability of a site type, t, as follows:

$$\begin{aligned} P(t|data) &= \frac{P(data|t)P(t)}{P(data)} \\ &= \frac{P(data|t)P(t)}{\sum_{s \in S} P(data|s)P(s)}, \end{aligned} \tag{22.11}$$

where $P(t)$ is the prior probability of site type t (that is, $f_s, s \in S$, from above) and $P(data|t)$ is the likelihood of the data, given the site type t. The data consists of all the nucleotides in all the reads in the mapping. For a particular site, assume that we have k reads that cover this site, and let i be an index over the nucleotides observed, n_i, in the reads at this site. We thus have:

$$P(data|t) = P(n_1, ..., n_k|t).$$

To derive the likelihood of the data, $P(n_1, ..., n_k|t)$, we first need some notation: For a given site type, t, let $P_t(N)$ be the probability that an allele from this site type has the nucleotide N. The $P_t(N)$ probabilities are known and are determined by the ploidy: For a diploid organism, $P_t(N) = 1$ if t is a homozygous site and N is one of the alleles in t, whereas it is 0.5 if t is a heterozygous and N is one of the alleles in t, and it is 0, if N is *not* one of the alleles in t. For a triploid organism, the $P_t(N)$ will be either 0, 1/3, 2/3 or 1.

With this definition, we can write the likelihood of the data $n_1, ..., n_k$ in a site t as:

$$P(n_1, ..., n_k|t) = \prod_{i=1}^{k} \sum_{N \in \{A,C,G,T,-\}} P_t(N) \times e_q(N \to n_i). \tag{22.12}$$

Inserting this expression for the likelihood, and the prior site type frequencies f_s and f_t for $P(s)$ and $P(t)$, in the expression for the posterior probability (22.11), we thus have the following equation for the posterior probabilities of the site types:

$$\begin{aligned} P(t|n_1,...,n_k) &= \frac{P(n_1,...,n_k|t)f_t}{\sum_{s\in S} P(n_1,...,n_k|s)f_s} \\ &= \frac{\prod_{i=1}^{k}\sum_{N\in\{A,C,G,T,-\}} P_t(N)\times e_q(N\rightarrow n_i)f_t}{\sum_{s\in S}\prod_{i=1}^{k}\sum_{N\in\{A,C,G,T,-\}} P_s(N)\times e_q(N\rightarrow n_i)f_s} \end{aligned} \quad (22.13)$$

The unknowns in this equation are the prior site type probabilities, $f_s, s \in S$, and the error rates $\{e(N \rightarrow M)|N, M \in \{A, C, G, T, -\}\}$. Once these have been estimated, we can calculate the posterior site type probabilities using the equation 22.13 for each site type, and hence, for each site, evaluate whether the sum of the posterior probabilities of the non-homozygous reference site types is larger than the cut-off. If so, we will set out current estimated site type to be that with the highest posterior probability.

Estimating the parameters in the model for the Fixed Ploidy Variant Caller

The Fixed Ploidy Variant Caller uses the Expectation Maximization (EM) procedure for estimating the unknown parameters in the model, that is, the prior site type probabilities, $f_s, s \in S$ and the error rates $\{e(N \rightarrow M)|N, M \in \{A, C, G, T, -\}\}$. The EM procedure is an iterative procedure: it starts with a set of initial prior site type frequencies, $f_s^0, s \in S$ and a set of initial error probabilities, $\{e_q^0(N \rightarrow M)|N, M \in \{A, C, G, T, -\}\}$. It then iteratively updates first the prior site type frequencies (to get $f_s^1, s \in S$), then the error probabilities (to get $\{e_q^1(N \rightarrow M)|N, M \in \{A, C, G, T, -\}\}$), then the site type frequencies again, etc. (a total of four rounds), in such a manner that the observed nucleotide patterns at the sites in the alignment become increasingly likely. To give an example of the forces at play in this iteration: as you increase the error rates you will decrease the likelihood of observing 'clean' patterns (e.g. patterns of only As and Cs at site types A/C) and increase the likelihood of observing 'noisy' patterns (e.g. patterns of other than only As, and C at site types A/C). If, on the other hand, you decrease the error rates, you will increase the likelihood of observing 'clean' patterns and decrease the likelihood of observing 'noisy' patterns. The EM procedure ensures that the balance between these two is optimized relative to the data observed (assuming, of course, that the ploidy assumption is valid).

Updating equations for the prior site type probabilities

We first derive the updating equations for the prior site type probabilities $f_s, s \in S$. The probability that the site is of type t given that we observe the nucleotides $n_1, ..., n_k$ in the reads at the site is:

$$\begin{aligned} P(t|n_1,...,n_k) &= \frac{P(t,n_1,...,n_k)}{\sum_{s\in S} P(s,n_1,...,n_k)} \\ &= \frac{P(t)P(n_1,...,n_k|t)}{\sum_{s\in S} P(s)P(n_1,...,n_k|s)} \end{aligned} \quad (22.14)$$

Now, for $P(t)$ we use our current value for f_t, and if we further insert the expression for $P(n_1, ..., n_k|t)$ (22.12) we get:

$$P(t|n_1, ..., n_k) = \frac{f_t \prod_{i=1}^{k} \sum_{N \in \{A,C,G,T,-\}} P_t(N) \times e_q(N \to n_i)}{\sum_{s \in S} f_s \prod_{i=1}^{k} \sum_{N \in \{A,C,G,T,-\}} P_s(N) \times e_q(N \to n_i)} \tag{22.15}$$

We get the updating equation for the prior site type probabilities, $f_t, t \in S$, from equation 22.15: Let h index the sites in the alignment ($h = 1, ...H$). Given the current values for the set of site frequencies, $f_t, t \in S$, and the current values for the set of error probabilities, we obtain updated values for the site frequencies, $f_t^*, t \in S$, by summing the site type probabilities given the data (as given by equation 22.15) across all sites in the alignment:

$$f_t^* = \frac{\sum_{h=1}^{H} \frac{f_t \prod_{i=1}^{k} \sum_{N \in \{A,C,G,T,-\}} P_t(N) \times e_q(N \to n_i^h)}{\sum_{s \in S} f_s \prod_{i=1}^{k} \sum_{N \in \{A,C,G,T,-\}} P_s(N) \times e_q(N \to n_i^h)}}{H} \tag{22.16}$$

Updating equations for the error rates

For the updating equations for the error probabilities, we consider a read, i, at a given site, h. The joint probability of the true nucleotide in the read, r_i^h, at the site being N and the data $n_1^h, ..., n_{k_h}^h$ is:

$$\begin{aligned}
&P(r_i^h = N, n_1^h, ..., n_{k_h}^h) \\
&= \sum_{s \in S} f_s P(r_i^h = N, n_1^h, ..., n_{k_h}^h | s) \\
&= \sum_{s \in S} f_s \prod_i P(r_i^h = N, n_i^h | s) \\
&= \sum_{s \in S} f_s P(r_i^h = N, n_i^h | s) \prod_{j \neq i} P(n_j^h | s) \\
&= \sum_{s \in S} f_s (P_s(N) \times e_{q_{ih}}(N \to n_i^h) \prod_{j \neq i} \sum_{N' \in \{A,C,G,T,-\}} P_s(N') \times e_{q_j}(N' \to n_j^h))
\end{aligned} \tag{22.17}$$

Using Bayes formula again, as we did above in 22.14, we get:

$$\begin{aligned}
P(r_i^h = N | n_1^h, ..., n_{k_h}^h) &= \frac{P(r_i^h = N, n_1^h, ..., n_k^h)}{P(n_1^h, ..., n_{k_h}^h)} \\
&= \frac{P(r_i^h = N, n_1^h, ..., n_{k_h}^h)}{\sum_{N' \in \{A,C,G,T,-\}} P(r_i^h = N', n_1^h, ..., n_{k_h}^h)}
\end{aligned} \tag{22.18}$$

and inserting the expression from equation 22.17:

$$
\begin{aligned}
&P(r_i^h = N|n_1^h, ..., n_{k_h}^h) \\
&= \frac{\sum_{s\in S} f(s)(P_s(N) \times e_{q_{i^h}}(N \to n_i^h) \prod_{j\neq i} \sum_{N'\in\{A,C,G,T,-\}} P_s(N') \times e_{q_j^h}(N' \to n_j^h))}{\sum_{N'\in\{A,C,G,T,-\}}(\sum_{s\in S} f(s)(P_s(N') \times e_{q_{i^h}}(N' \to n_i^h) \prod_{j\neq i} \sum_{N'\in\{A,C,G,T,-\}} P_s(N') \times e_{q_j^h}(N' \to n_j^h)))} \\
&= \frac{\sum_{s\in S} f(s)(P_s(N) e_{q_{i^h}}(N \to n_i^h) \prod_{j\neq i} \sum_{N'\in\{A,C,G,T,-\}} P_s(N') \times e_{q_j^h}(N' \to n_j^h))}{(\sum_{s\in S} f(s) \prod_j \sum_{N'\in\{A,C,G,T,-\}} P_s(N') \times e_{q_j^h}(N' \to n_j^h)} \qquad (22.19)
\end{aligned}
$$

The equation 22.19 gives us the probabilities for a given read, i, and site, h, given the data $n_1^h, ..., n_k^h$, that the true nucleotide is N, $N \in \{A, C, G, T, -\}$, given our current values of the error rates and site probabilities. Since we know the sequenced nucleotide in each read at each site, we can get new updated values for the error rate of producing an M nucleotide when the true nucleotide is N, $e_q^*(N \to M)$, for $N, M \in \{A, C, G, T, -\}$ by summing the probabilities of the true nucleotide being N for all reads across all sites for which the sequenced nucleotide is M, and dividing by the sum of all probabilities of the true nucleotide being a N across all reads and all sites:

$$
e_q^*(N \to M) = \frac{\sum_h \sum_{i=1,...,k_h : n_i^h = M} P(r_i^k = N|n_1^h, ..., n_{k_h}^h)}{\sum_h \sum_{i=1,...,k_h} P(r_i^k = N|n_1^h, ..., n_{k_h}^h)}
$$

22.19.2 The Low Frequency Variant caller: Models and methods

This section describes the model, method and estimation procedure behind the Low Frequency Variant Caller. The Low Frequency Variant Caller is designed to detect variants in a sample for which the ploidy is unknown. The Low Frequency Variant Caller has a particularly high *sensitivity* for detecting variants that are present at any, and in particularly at low, allele frequencies.

The model for the Low Frequency Variant Caller

The purpose of the Low Frequency Variant Caller is to call variants in samples with unknown ploidy (e.g. samples of limited tumor purity from cancer patients) using read mapping data. It will detect germline as well as somatic variants, and may also be used on samples from other high-ploidy organisms, or pooled samples. Like the Fixed Ploidy Variant Caller it detects Single Nucleotide Variants (SNVs), MNVs (Multiple Nucleotide Variants), insertions, deletions as well as replacements (combinations of neighboring insertions, deletions and SNVs for which the positions are ambiguous).

The algorithm behind the The Low Frequency Variant Caller relies on Multinomial models for the presence of different nucleotide alleles at a given site and an error model for the sequencing (the error model is identical to that of the Fixed Ploidy variant caller). The Multinomial models are of the kind "there are q different nucleotide alleles present at the site with frequencies f_i, $i = 1, ..., q$, $\sum_{i=1}^{q} f_i = 1$", where the number of alleles, q, differ. The models that are evaluated at each site are given in Table 22.1.

In words, model M_x can be described as: "There is really only the X nucleotide allele present at the site, *all other nucleotides are due to errors*" and model $M_{x,y,z}$ as: "There are really only the nucleotide alleles X, Y and Z present at the site, *all other nucleotides are due to errors*". The

Model	**Alleles present at the site**	**Description**	**Free parameters***
M_x	x	the only allele present at the site is x.	none
$M_{x,y}$	x and y	x is present at frequency $1-f$, y at frequency f	f
$M_{x,y,z}$	x, y and z	x is present at frequency $1-(f_1+f_2)$, y at frequency f_1 and z at frequency f_2	f_1 and f_2
$M_{x,y,z,w}$	x, y, z and w	x is present at frequency $1-(f_1+f_2+f_3)$, y at frequency f_1, z at frequency f_2 and w at frequency f_3	f_1, f_2 and f_3
$M_{x,y,z,w,v}$	x, y, z, w and v	x is present at frequency $1-(f_1+f_2+f_3)$, y at frequency f_1, z at frequency f_2, w at frequency f_3 and w at frequency f_4	f_1, f_2, f_3 and f_4

Table 22.1: The Multinomial models evaluated at each site. X, Y, Z, W and V each take on one of the values A, C, G, T, or$-$ ($X \neq Y \neq Z \neq W \neq V$). Free parameters*: the parameters that are free in each of the Multinomial models of the Low Frequency Variant Caller.

hypotheses where a single nucleotide is present in the sample have no free parameters (there is just one frequency parameter and it must be 1). A hypothesis stating that a site is a mixture of two different nucleotides, e.g. [A/G] has one free parameter since there are frequencies for two nucleotides but they have to sum to one.

Parameter estimation relies on the Maximum Likelihood principle, and, as the Fixed Ploidy Variant Caller, the EM algorithm is used for estimating the parameters of the model. Given an initial set of parameter values for the error rates, the different Multinomial models are evaluated at each site by finding the maximum likelihood estimates of the frequency parameters for each model. The model that offers the best explanation of the data (while taking care to adjust for the numbers of parameters in the Multinomial model, using a criterion adopted from the Akaike Information criterion) is chosen as the current guess of the true allelic situation at that site, and given that, the error rates are re-estimated. Given the new error estimates, the Maximum Log Likelihoods for all possible Multinomial models are again evaluated and updated frequencies are produced. This procedure is performed a total of four times. After the final round of estimation the Multinomial model that offers the best explanation of the data is chosen as the winning model, and variants are called according to that model.

Below we describe in detail how we choose among competing models and derive the updating equations for the EM estimation of the frequency and error rate parameters.

Updating the choice of favored Multinomial model for each site

Given a set of error rates, and, for each site, a set of Maximum Likelihood estimates of the nucleotide allele frequencies for each Multinomial model, we can calculate the Maximum loglikelihood values for each of the models at each site. Since the hypotheses with fewer free parameters are special cases of hypotheses with more free parameters, the hypotheses with

the most free parameters will necessarily have the highest likelihoods. We wish to only favor a model with more parameters over one with fewer, if it offers a significantly better explanation of the data at the site. In the simple case where we have nested models (e.g. a hypothesis, H_0, with no free parameters and an alternative hypothesis, H_1, which has one free parameter and contains H_0 as a special case) it is a well known result that twice the loglikehood ratio is approximately χ^2 distributed with a parameter that is equal to the difference between the number of free parameters in the hypotheses, n:

$$2 \log \frac{L(H_1)}{L(H_0)} \sim \chi^2(n).$$

If we write $c_n(p)$ for the inverse cumulative probability density function for a $\chi^2(n)$ distribution evaluated at $1-p$, we get a cutoff value for when we prefer H_1 over H_0 at the significance level given by p.

We wish to compare all models together and some of these will not be nested. We therefore generalize this approach to apply to any set of Multinomial model hypothesis H_m, $m = 1..., M$. For each model we calculate the value:

$$v_m = 2 \log L(H_m) - c_{df_m}(p) \quad (22.20)$$

where df_m is the number of free parameters in hypothesis H_m. We now prefer the hypothesis with the highest value of v (note that when comparing a hypothesis with zero free parameters to another hypothesis, we get exactly the same results as with the log likelihood ratio approach). If this hypothesis is that only the reference allele is present at the site, no variants are called. For other models, the variants corresponding to the alleles hypothesized by the models are called.

The approach applied is an extension of the Akaike approach: In the Akaike approach you subtract twice the number of parameters, whereas in the approach that we apply, we subtract $c_{df_x}(p)$, whereby additional parameters are punished relative to the significance level, p, rather than in a fashion that is linear in the number of parameters. This means that the more parameters that are present, the more reluctant we will be to add even more.

Updating equations for the Multinomial model frequency parameters

Consider a site, h, and let ${n_i}^h$ be the nucleotide observed in read i at this site, $i = 1, ..., k$. For each of the Multinomial models that may explain the data at the site we have a number of frequency parameters. For simplicity, we consider the model which states that there are two alleles present at the site, the reference allele, y, and another allele x, and let f be the frequency parameter for the non-reference allele (hence the frequency of the reference allele, f_y, is $1-f$). Models with more alleles are treated in a similar manner.

We want to estimate the parameter for the frequency of the x allele at the site h, f, by the fraction of true nucleotides that are x at this site,given the observed data:

$$f^* = \frac{\sum_{i=1}^{k} P(r_i^h = x | n_i^h)}{k}. \quad (22.21)$$

To calculate this we use Bayes Theorem on the numerator:

$$P(r_i^h = x|n_i^h) = \frac{P(r_i^h = x, n_i^h)}{P(n_i^h)} \quad (22.22)$$

$$= \frac{P(x) \times e(x \to n_i^h)}{P(x) \times e(x \to n_i^h) + P(y) \times e(y \to n_i^h)} \quad (22.23)$$

$$= \frac{f \times e(x \to n_i^h)}{f \times e(x \to n_i^h) + (1-f) \times e(y \to n_i^h)} \quad (22.24)$$

Inserting our current values for the frequency parameter f under the model, and the error rates $e(x \to n_i^h)$ and $e(y \to n_i^h)$, in 22.22, and further inserting the obtained values in 22.21 gives us updated values for the frequency parameter f.

Updating equations for the error rates

Consider a site h and a read i. The joint probability of the true nucleotide in the read, r_i^h, at the site being N and the observed nucleotide at the site n_i^h is:

$$P(r_i^h = N, n_i^h) = P_h(N) e_{q_i^h}(N \to n_i^h). \quad (22.25)$$

Using Bayes Theorem, the probability of the true nucleotide in the read, r_i^h, at the site being N, given that we observe n_i^h is:

$$P(r_i^h = N|n_i^h) = \frac{P(r_i^h = N, n_i^h)}{\sum_{N' \in A,C,G,T,-} P(r_i^h = N', n_i^h)}. \quad (22.26)$$

Inserting 22.25 in 22.26 we get:

$$P(r_i^h = N|n_i^h) = \frac{P_h(N) e_{q_i^h}(N \to n_i^h)}{\sum_{N' \in A,C,G,T,-} P_h(N') e_{q_i^h}(N' \to n_i^h)}. \quad (22.27)$$

The equation 22.27 gives us the probabilities for a given read, i, and site, h, given the observed nucleotide n_i^h, that the true nucleotide is N, $N \in \{A, C, G, T, -\}$, given our current values for the frequency f (inserted for $P_h(N)$) and error rates. Since we know the sequenced nucleotide in each read at each site, we can get new updated values for the error rate of producing an M nucleotide when the true nucleotide is N, $e_q^*(N \to M)$, for $N, M \in \{A, C, G, T, -\}$ by summing the probabilities of the true nucleotide being N for all reads across all sites for which the sequenced nucleotide is M, and dividing by the sum of all probabilities of the true nucleotide being a N across all reads and all sites:

$$e_q^*(N \to M) = \frac{\sum_h \sum_{i=1,\ldots,k_h : n_i^h = M} P(r_i^k = N|n_i^h)}{\sum_h \sum_{i=1,\ldots,k_h} P(r_i^k = N|n_i^h)}$$

Probability of the variant

A probability value is reported for each variant call in the variant track table column with the header "Probability". For the Fixed Ploidy Variant Caller this value is the posterior probability of

the site being different from the homozygous reference site. This is possible to calculate because the model behind the Fixed Ploidy Variant Caller is Bayesian. As the model for the Low Frequency Variant Caller is not a Bayesian model, such a value strictly speaking does not exist. However, as a proxy for such a value we report the support for the winning model m relative to the total support of all models:

$$P_m = \frac{e^{v_m}}{\sum_{m'} e^{v_{m'}}}$$

22.20 Variant data

Variant data may be obtained either by importing variants from files (e.g. gvf or vcf files - as described in section 6.2), by downloading variants from external databases (e.g. dbSNP, HapMap, or 1000genomes - (described in section 6.2)) or by calling variants on read tracks or read mappings using the CLC Basic Variant Detection (section 22.15), Fixed Ploidy Variant Detection (section 22.13), or the Low Frequency Variant Detection (section 22.14) tools.

Variant types include SNVs, MNVs, insertions, deletions or replacements. They may be presented either in a variant track (see figure 22.88) or in an annotated variant table (see figure 22.91).

22.20.1 Variant tracks

A variant track (figure 22.88), created with the *Biomedical Genomics Workbench* variant callers (see section 22.12), has the following information for each variant:

Chromosome The name of the reference sequence on which the variant is located.

Region The region on the reference sequence at which the variant is located. The region may be either a 'single position', a 'region' or a 'between position region'. Examples are given in figure 22.89. An extract of a gvf-file giving rise to these three variants after import is shown in figure 22.90.

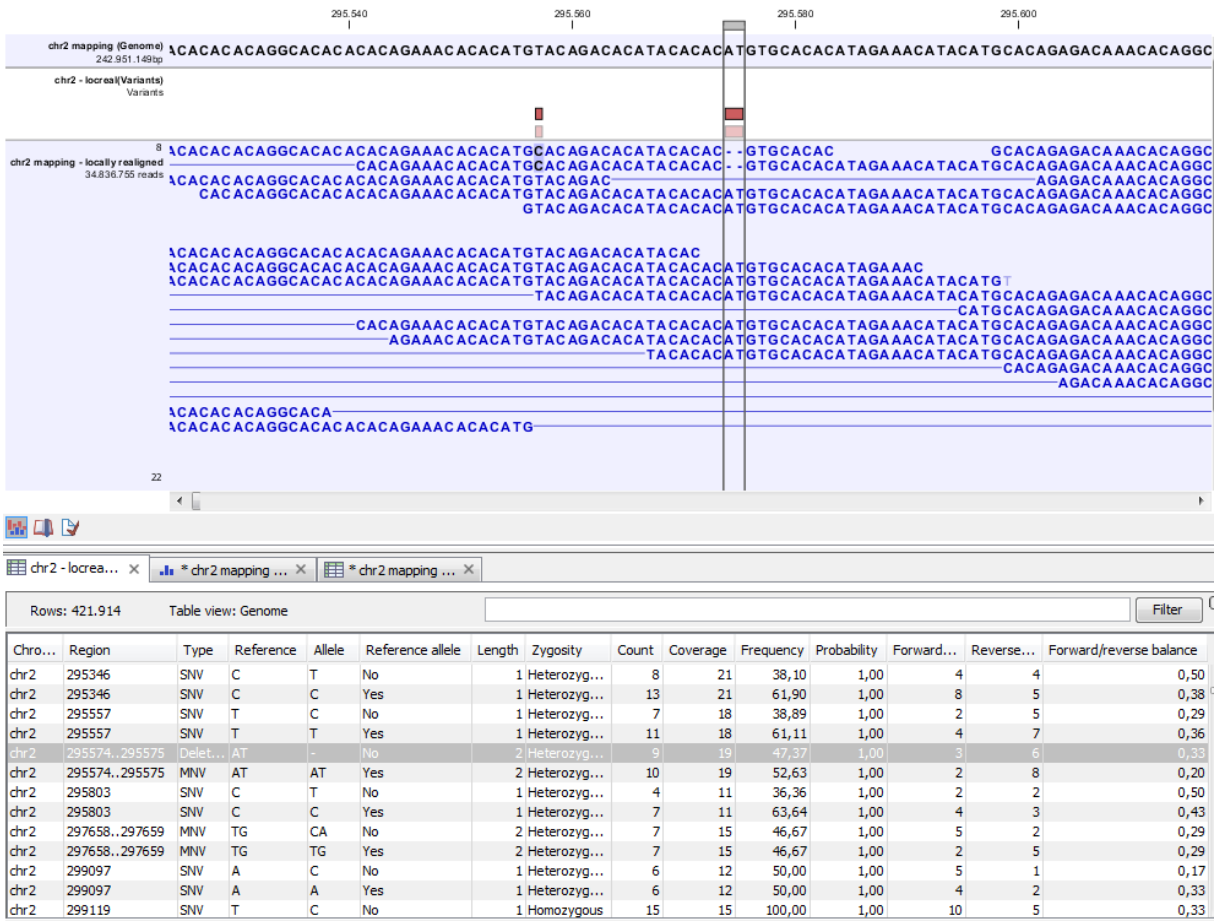
Variant type The type of variant. This can either be SNV (single-nucleotide variant), MNV (multi-nucleotide variant), insertion, deletion, or replacement. Learn more in section 22.20.3.

Reference The reference sequence at the position of the variant.

Allele The allele sequence of the variant.

Reference allele Describes whether the variant is identical to the reference. This will be the case one of the alleles for most, but not all, detected heterozygous variants (e.g. the variant caller might detect two variants, A and G, at a given position in which the reference is 'A'. In this case the variant corresponding to allele 'A' will have 'Yes' in the 'reference allele' column entry, and the variant corresponding to allele 'G' would have 'No'. Had the variant caller called the two variants 'C' and 'G' at the position, both would have had 'No' in the 'Reference allele' column).

Length The length of the variant. The length is 1 for SNVs, and for MNVs it is the number of allele or reference bases (which will always be the same). For deletions, it is the length of the deleted sequence, and for insertions it is the length of the inserted sequence. For



Chro...	Region	Type	Reference	Allele	Reference allele	Length	Zygosity	Count	Coverage	Frequency	Probability	Forward...	Reverse...	Forward/reverse balance
chr2	295346	SNV	C	T	No	1	Heterozyg...	8	21	38,10	1,00	4	4	0,50
chr2	295346	SNV	C	C	Yes	1	Heterozyg...	13	21	61,90	1,00	8	5	0,38
chr2	295557	SNV	T	C	No	1	Heterozyg...	7	18	38,89	1,00	2	5	0,29
chr2	295557	SNV	T	T	Yes	1	Heterozyg...	11	18	61,11	1,00	4	7	0,36
chr2	295574..295575	Delet...	AT	-	No	2	Heterozyg...	9	19	47,37	1,00	3	6	0,33
chr2	295574..295575	MNV	AT	AT	Yes	2	Heterozyg...	10	19	52,63	1,00	2	8	0,20
chr2	295803	SNV	C	T	No	1	Heterozyg...	4	11	36,36	1,00	2	2	0,50
chr2	295803	SNV	C	C	Yes	1	Heterozyg...	7	11	63,64	1,00	4	3	0,43
chr2	297658..297659	MNV	TG	CA	No	2	Heterozyg...	7	15	46,67	1,00	5	2	0,29
chr2	297658..297659	MNV	TG	TG	Yes	2	Heterozyg...	7	15	46,67	1,00	2	5	0,29
chr2	299097	SNV	A	C	No	1	Heterozyg...	6	12	50,00	1,00	5	1	0,17
chr2	299097	SNV	A	A	Yes	1	Heterozyg...	6	12	50,00	1,00	4	2	0,33
chr2	299119	SNV	T	C	No	1	Homozygous	15	15	100,00	1,00	10	5	0,33

Figure 22.88: *Variant track. The figure shows a track list (top), consisting of a reference sequence track, a variant track and a read mapping. The variant track was produced by running the Fixed Ploidy Variant Detection tool on the read track. The variant track has been opened in a separate table view by double-clicking on it in the track list. By selecting a row in the variant track table, the track list view is centered on the corresponding variant.*

replacements, both the length of the replaced reference sequence and the length of the inserted sequence are considered, and the longest of those two is reported.

Zygosity The zygosity of the variant called, as determined by the variant caller. This will be either 'Homozygous', where there is only one variant called at that position or 'Heterozygous' where more than one variant was called at that position.

Count The number of 'countable' reads supporting the allele. The 'countable' reads are those that are used by the variant caller when calling the variant. Which reads are 'countable' depends on the user settings when the variant calling is performed - if e.g. the user has chosen 'Ignore broken pairs', reads belonging to broken pairs are not 'countable'.

Coverage The read coverage at this position. Only 'countable' reads are considered (see under 'Count' above for an explanation of 'countable' reads. Also see overlapping pairs in section 22.21 for how overlapping paired reads are treated.)

Frequency The number of 'countable' reads supporting the allele divided by the number of 'countable' reads covering the position of the variant ('see under 'Count' above for an

explanation of 'countable' reads).

Probability The probability that this particular variant exists in the sample. (For further information please refer to the White paper on Probabilistic Variant Caller: http://resources.qiagenbioinformatics.com//white-papers/White_paper_on_probabilistic_variant_caller_1.1.pdf).

Forward read count The number of 'countable' forward reads supporting the allele (see under 'Count' above for an explanation of 'countable' reads). Also see more information about overlapping pairs in section 22.21.

Reverse read count The number of 'countable' reverse reads supporting the allele (see under 'Count' above for an explanation of 'countable' reads). Also see more information about overlapping pairs in section 22.21.

Forward/reverse balance The minimum of the fraction of 'countable' forward reads and 'countable' reverse reads carrying the variant among all 'countable' reads carrying the variant (see under 'Count' above for an explanation of 'countable' reads).[2]

Average quality The average read quality score of the bases supporting a variant. If there are no values in this column, it is probably because the sequencing data was imported without quality scores (learn more about importing quality scores from different sequencing platforms in section 6.3). For deletions, the quality scores of the two surrounding bases are taken into account, and the lowest value of these two is reported.

Hyper-allelic Basic and Fixed Ploidy Variant detectors only: contains "yes", if the site contains more variants than the user-specified ploidy predicts, "no" if not.

QUAL This value is necessary for certain downstream analyses of the data after export in vcf format. It is calculated as

$$-10\log_{10}(1-p) \tag{22.28}$$

p being the probability that a particular variant exists in the sample (see above for the definition of probability). A QUAL value of 10 indicates a 1 in 10 chance that the called variant is an error, while a QUAL of 100 indicates a 1 in 10^{10} chance that the called variant is an error. QUAL is capped at 200 for p=1.

Please note that the variants in the variant track can be enriched with information using the annotation tools in section 23.

A variant track can be imported and exported in VCF or GVF formats. An example of the gvf-file giving rise to the variants shown in figure 22.89 is given in figure 22.90.

[2] Some systematic sequencing errors can be triggered by a certain combination of bases. This means that sequencing one strand may lead to sequencing errors that are not seen when sequencing the other strand (see [Nguyen et al., 2011] for a recent study with Illumina data). In order to evaluate whether the distribution of forward and reverse reads is approximately random, this value is calculated as the minimum of the number of forward reads divided by the total number of reads and the number of reverse reads divided by the total number of reads supporting the variant. An equal distribution of forward and reverse reads for a given allele would give a value of 0.5. (See also more information about overlapping pairs in section 22.21.)

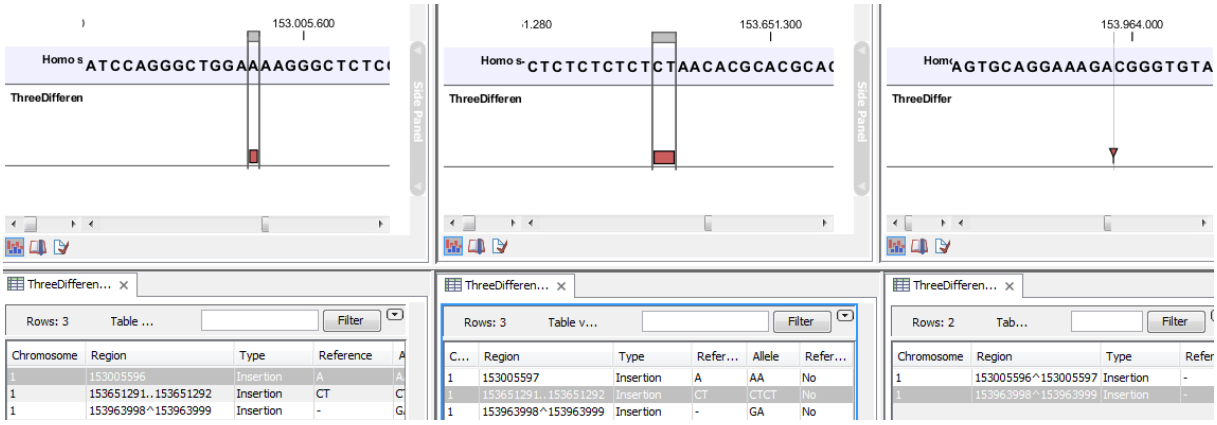


Figure 22.89: *Examples of variants with different types of 'Region' column contents. The left-most variant has a 'single position' region, the middle variant has a 'region' region and the right-most has a 'between positions' region.*

```
##gff-version 3
##gvf-version 1.06
##file-date 2013-09-23
#file-encoding windows-1252
1   CLC insertion   153005596   153005596   0   .   .   ID=CLC_1;Variant_seq=AA;Reference_seq=A;
1   CLC insertion   153651291   153651292   0   .   .   ID=CLC_2;Variant_seq=CTCT;Reference_seq=CT;
1   CLC insertion   153963999   153963998   0   .   .   ID=CLC_3;Variant_seq=GA;Reference_seq=-;
```

Figure 22.90: *A gvf file giving rise to the variants in the figure above.*

22.20.2 The annotated variant table

While the track table contains reference alleles and non-reference alleles, the annotated table lists only non-reference alleles. It include for each allele a subset of the columns of the variant track table and three additional columns (see figure 22.91).

Reference...	Type	Reference	Allele	Overlapping annotations	Coding region change	Amino acid change
3574524	SNV	T	C			
3574532	SNV	T	C			
3574536	SNV	T	C			
3575808	SNV	A	T	Gene: TEP1, mRNA: TEP1		
3655632	SNV	C	A	Gene: OSGEP, CDS: OSGEP, mRNA: OSGEP	NP_060277.1:c.681G>T	NP_060277.1:p.Glu227Asp
3655679	Deletion	A	-	Gene: OSGEP	NP_060277.1:c.637-3delT	
3655684	SNV	T	G	Gene: OSGEP	NP_060277.1:c.637-8A>C	
3656277	SNV	C	T	Gene: OSGEP, CDS: OSGEP, mRNA: OSGEP	NP_060277.1:c.597G>A	
3656304	SNV	T	C	Gene: OSGEP, CDS: OSGEP, mRNA: OSGEP	NP_060277.1:c.570A>G	

Figure 22.91: *An example of an annotated variant table.*

When the variant calling is performed on a read mapping in which gene and cds annotations are present on the reference sequence, the three columns will contain the following information:

Overlapping annotation This shows if the variant is covered by an annotation. The annotation's type and name will displayed. For annotated reference sequences, this information can be used to tell if the variant is found in a coding or non-coding region of the genome. **Note** that annotations of type `Variation` and `Source` are not reported.

Coding region change For variants that fall within a coding region of a gene, the change is reported according to the standard conventions as outlined in `http://varnomen.hgvs.org/`.

Amino acid change If the reference sequence of the mapping is annotated with ORF or CDS

annotations, the variant caller will also report whether the variant is synonymous or non-synonymous. If the variant changes the amino acid in the protein translation, the new amino acid will be reported. The nomenclature used for reporting is taken from http://varnomen.hgvs.org/.

If the reference sequence has no gene and cds annotations these columns will have the entry "NA".

The table can be **Exported** () as a csv file (comma-separated values) and imported into e.g. Excel. Note that the CSV export includes all the information in the table, regardless of filtering and what has been chosen in the **Side Panel**. If you only want to use a subset of the information, simply select and **Copy** () the information.

Note that if you make a split view of the table and the mapping (see section 2.1.6), you will be able to browse through the variants by clicking in the table. This will cause the view to jump to the position of the variant.

This table view is not well-suited for downstream analysis, in which case we recommend working with tracks instead (see section 22.20.1).

22.20.3 Variant types

Variants are classified into five different types:

SNV A single nucleotide variant. This means that one base is replaced by one other base. This is also often referred to as a SNP. *SNV* is preferred over *SNP* because the latter includes an extra layer of interpretation about variants in a population. This means that an SNV could potentially be a SNP but this cannot be determined at the point where the variant is detected in a single sample.

MNV This type represents two or more SNVs in succession.

Insertion This refers to the event where one or more bases are inserted in the experimental data compared to the reference.

Deletion This refers to the event where one or more bases are deleted from the experimental data compared to the reference.

Replacement This is a more complex event where one or more bases have been replaced by one or more bases, where the identified allele has a length different from the reference (i.e. involving an insertion or deletion). Basically, this type represents variants that cannot be represented in the other four categories. An example could be `AAA->CC`. This cannot be resolved into a SNV or an MNV because the number of bases is different between the experimental data and the reference, it is not an insertion because something is also deleted from the reference, and it is not a deletion because something is also inserted.

Note about overlapping variants: If two different types of variants occur in the same location, these are reported separately in the output table. This is particularly important when SNPs occur in the same position as an MNV. Usually, multiple SNVs occurring alongside each other would simply be reported as one MNV, but if one SNV of the MNV is found in additional case samples by itself, it will be reported separately. For example, if an MNV of AAT -> GCA at position 1 occurs

in five of the case samples, and the SNV at position 1 of A -> G occurs in an additional 3 samples (so 8 samples in total), the output table will list the MNV and SNV information separately. However, the SNV will be shown as being present in only 3 samples, as this is the number in which it appears "alone".

22.21 Detailed information about overlapping paired reads

Paired reads that overlap introduce additional complexity for variant detection. This section describes how this is handled by *Biomedical Genomics Workbench*.

When it comes to **coverage** in the overlapping region, each pair is contributing once to the coverage. Even if there are indeed two reads in this region, they do not both contribute to coverage. The reason is that the two reads represent the same fragment, so they are essentially treated as one.

When it comes to counting the number of **forward and reverse reads**, including the forward/reverse reads balance, each read contribute. This is because this information is intended to account for systematic sequencing errors in one direction, and the fact that the two reads are from the same fragment is less important than the fact that they are sequenced on different strands.

If the two overlapping reads do not agree about the variant base, they are both ignored. Please note that there can be a special situation with the basic variant detection: If the two reads disagree, and one read does not pass the quality filter, the other read will contribute to the variant just as if there had been only that read and no overlapping pair.

22.22 Identify Known Mutations from Sample Mappings

The **Identify Known Mutations from Sample Mappings** tool can be used to look up known genomic variants in read mappings. This can be done in one or more samples by comparing a track of known variants with the read mappings of interest in order to test for the presence or absence of relevant variants in patient samples for example.

The **Identify Known Mutations from Sample Mappings** tool does not perform any kind of variant calling, which means that this tool cannot be used to find de novo variants. Rather, the tool is intended for identification of variants that have already been reported.

You need two types of input for the **Identify Known Mutations from Sample Mappings** tool:

- A variant track that holds the specific variants that you wish to test for.
- The read mapping(s) that you wish to check for the presence (or absence) of specific variants.

The **Identify Known Mutations from Sample Mappings** tool has two kinds of outputs:

- An overview track with information about:
  - whether the variant could be detected or not
  - whether the coverage was sufficient at the given position
  - the frequency of the variant in each sample.

- Individual output tracks for each sample that show the observed frequency, average base quality, forward/reverse read balance, zygosity and observed allele count.

22.22.1 How to run the Identify Known Mutations from Sample Mappings tool

To run the "Identify Known Mutations from Sample Mappings" tool go to the toolbox:

Toolbox | Resequencing Analysis ( ) | Identify Known Mutations from Sample Mappings ( )

This opens the wizard shown where you can specify the read mapping(s) to analyze. Click **Next** to get the following options (figure 22.92):

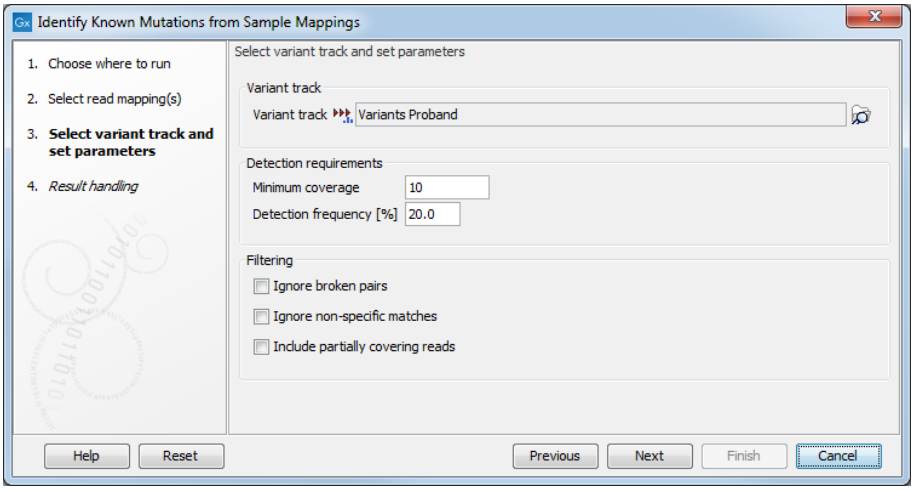


Figure 22.92: *Select the variant track with the variants that you wish to use for variant testing.*

Variant track

- **Variant track** Select the variant track that contains the specific variants that you wish to test for in your read mapping. **Note!** You can only select one variant track at the time. If you wish to compare with more than one variant track, you must run the analysis with each individual variant track at the time.

Detection requirements

- **Minimum coverage** The minimum number of reads that covers the position of the variant, which is required to set "Sufficient Coverage" to YES.
- **Detection frequency** The minimum allele frequency that is required to annotate a variant as being present in the sample. The same threshold will also be used to determine if a variant is homozygous or heterozygous. In case the most frequent alternative allele at the position of the considered variant has a frequency of less than this value, the zygosity of the considered variant will be reported as being homozygous.

Filtering

- **Ignore broken pairs** When ticked, reads from broken pairs are ignored. Broken pairs may arise for a number of reasons, one being erroneous mapping of the reads. In general, variants based on broken pair reads are likely to be less reliable, so ignoring them may reduce the number of spurious variants called. However, broken pairs may also arise for biological reasons (e.g. due to structural variants) and if they are ignored some true variants may go undetected.
- **Ignore non-specific matches** Reads that have an equally good match elsewhere on the reference genome (these reads are colored yellow in the mapping view) can be ignored in the analysis. Whether you include these reads or not will be a tradeoff between sensitivity and specificity. Including them may lead to the prediction of transcripts that are not correct, whereas excluding them may mean that you will loose some true transcripts.
- **Include partially covering reads** Reads that partially overlap variants (see the blue box below for a definition) will be considered to enable the detection of variants that are longer than the reads. When the "Include partially covering reads" option is disabled, only fully covering reads will be counted for all annotations. Enabling the "Include partially covering reads" option means that all fully covering reads will be counted for all annotations, and that additionally, partially covering reads will be included in relevant annotations including Coverage. Thus, if a partial read is compatible with multiple variants in the same region, the sum of all Counts for that region may be greater than the Coverage, and the sum of all Frequencies for that region may be higher than 100%.

A fully covering read is described as such:

- for SNV, MNV and Deletion: the read must cover all reference positions in the variant region.
- for Insertion and Replacement: the read must overlap adjacent reference positions on both sides of the variant region.

A partially covering read is read that is not fully covering the variant region, but overlaps with at least one residue.

Click on the button labeled **Next** to go to the next wizard step (figure 22.93). At this step the output options can be adjusted.

The output options are:

- **Create individual track** For each read mapping an individual track is created with the observed frequency, average base quality, forward/reverse read balance, zygosity and observed allele count.
- **Create overview track** The overview track is a summary for all samples with information about whether the coverage is sufficient at a given variant position and if the variant has been detected; the frequency of the variant.

Specify where to save the results and click on the button labeled **Finish**.

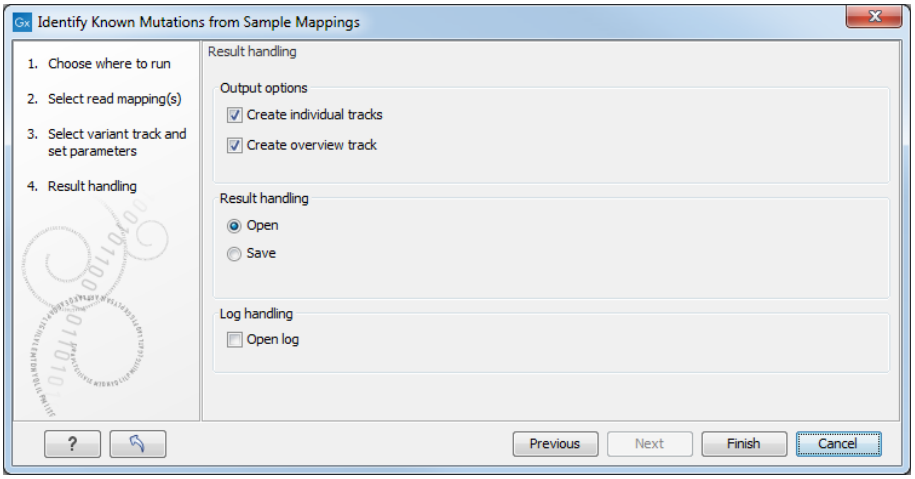


Figure 22.93: *Select the desired output format(s). If using the default settings, two types of output will be generated; individual tracks and overview tracks.*

22.22.2 Output from the Identify Known Mutations from Sample Mappings tool

One individual sample output track will be created for each read mapping analyzed, while one overview track will be created per analysis (figure 22.94).

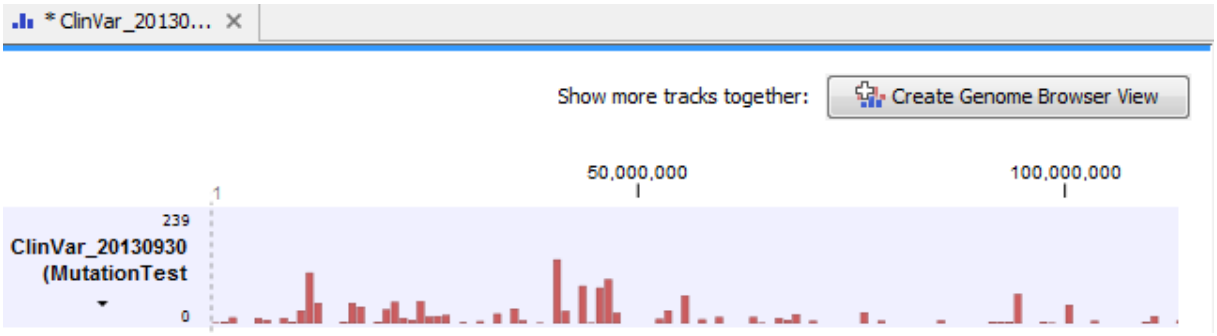


Figure 22.94: *Overview track of read mappings tested against a Clinvar variant track.*

At the bottom of the window it is possible to switch to a table view that lists all the mutations from the variant track that were found in your sample mapping.

In the individual track, the variant has been annotated with the following information:

- **Chromosome**
- **Region**
- **Type** SNV, MNV, Deletion, Insertion or Replacement
- **Reference**
- **Allele**
- **Reference allele** describes with a Yes or No whether the allele is similar to the reference.
- **Length** of the allele
- **Zygosity** Homozygous or Heterozygous (based on the parameter "Detection frequency" setting)

- **Count** Number of reads supporting the variant.
- **Coverage**
- **Frequency** Frequency of the reads supporting the variant
- **Forward read count**
- **Reverse read count**
- **Forward/Reverse balance** Minimum ratio of forward and reverse reads supporting the variant
- **Average Quality** Average quality of all bases supporting the variant
- **Most frequent alternative allele (MFAA)**
- **MFAA count** The Most Frequent Alternative Allele count (MFAA count) is the count of reads supporting the most frequent alternative allele at the position of the variant
- **MFAA frequency** Frequency of reads supporting the most frequent alternative allele at the position of the variant
- **MFAA forward read count** forward reads supporting the most frequent alternative allele at the position of the variant
- **MFAA reverse read count** reverse reads supporting the most frequent alternative allele at the position of the variant
- **MFAA forward/reverse balance** forward/reverse balance of the most frequent alternative allele at the position of the variant
- **MFAA average quality** average quality of the most frequent alternative allele at the position of the variant

In the overview track the variant has been annotated with the following information:

- **("Sample name") coverage** Either Yes or No, depending on whether the coverage at the position of the variant was higher or lower than the user given threshold for minimum coverage.
- **("Sample name") detection** Either Yes or No, depending on the minimum frequency settings chosen by the user.
- **("Sample name") frequency** The variant frequency observed in this sample.
- **("Sample name") zygosity** The zygosity observed in the sample. This setting is based on the minimum frequency setting made by the user. If this variant has been detected and the most frequent alternative allele at this position is also over the cutoff, the value is heterozygote.

An example of the individual and overview tables can be seen in figure 22.95.

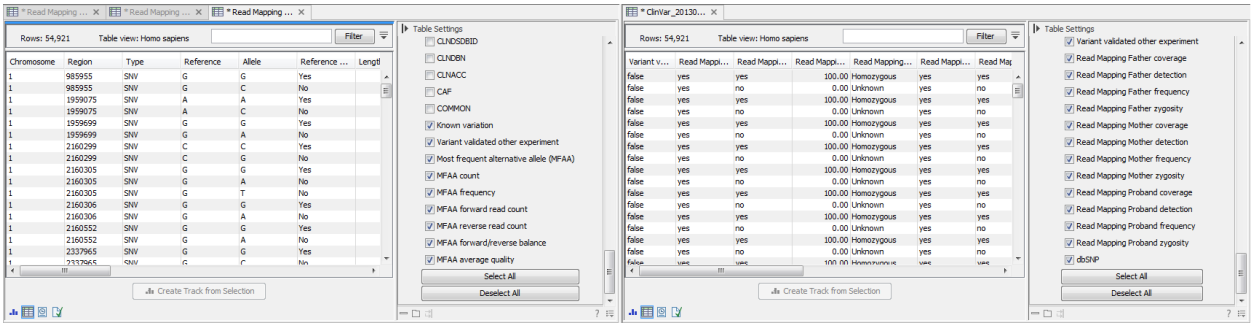


Figure 22.95: *Table views of the individual track (left) and overview track (right).*

Part VII

Working with variants

Chapter 23

Add information to variants tools

Contents

23.1 Add information from variant databases

It is also possible to annotate with information from variant databases. To run the Add information from variant databases tool, go to:

Toolbox | **Add Information to Variants** ( ) | **Add Information from Variant Databases** ( )

This tool will create a new track with all the experimental variants including added information about overlapping variants found in the track of known variants. The annotations are marked in three different ways:

Exact match This means that the variant position and allele both have to be identical in the input and the known variants track (however, note the extra option for joining adjacent SNVs and MNVs described below).

Partial MNV match This applies to MNVs which can be annotated with partial matches if an SNV or a shorter MNV in the database has an allele sequence that is contained in the allele sequence of the annotated MNV.

Overlap This will report if the known variant track has an overlapping variant.

For exact matches, all the information about the variant from the known variants track is transferred to the annotated variant. For partial matches and overlaps, the information from the known variants are not transferred.

23.2 Add conservation scores

The possible functional consequence of a variant can be interrogated by comparing to a conservation score that tells how conserved this particular position is among a set of different species. The underlying line of thought is that conserved bases are functionally important - otherwise they would have been mutated during evolution. If a variant is found at a position that is otherwise well conserved, it is an indication that the variant is functionally important. Of course this is only a prediction, as non-conserved regions could have functional roles too.

Conservation scores can be computed by several tools e.g. PhyloP and PhastCons and can be downloaded as pre-computed scores from an whole genome alignment of different species from different sources. See how to find and import tracks with conservation scores in section 6.2.

Toolbox | **Add Information to Variants** ( ) | **Add Conservation Scores** ( )

Select the variant track as input and when you click **Next** you will need to provide the track with conservation scores (see figure 23.1).

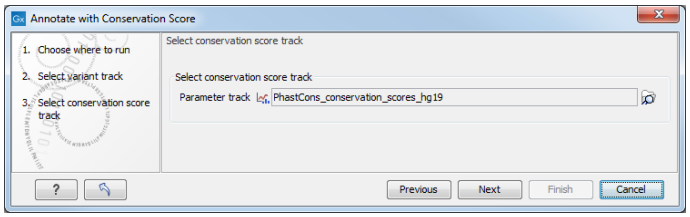


Figure 23.1: *The conservation score track.*

In the resulting track, all the variants will have quality scores annotated, and this can be used for sorting and filtering the track (see section 19.3.3).

23.3 Add exon number

Given a track with mRNA annotations, a new track will be created in which variants are annotated with the numbering of the corresponding exon with numbered exons based on the transcript

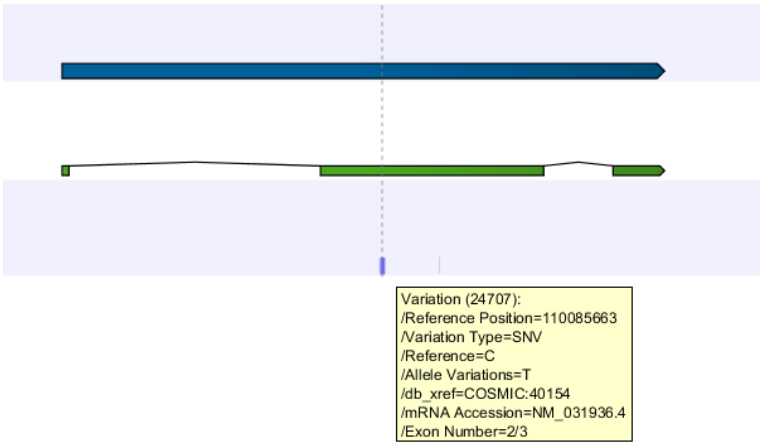


Figure 23.2: *A variant found in the second exon out of three in total.*

annotations in the input track (see an example of a result in figure 23.2).

When there are multiple isoforms, a comma-separated list of the exon numbers is given.

23.4 Add flanking sequence

In some situations, it is useful to see a variant in the context of the bases of the reference sequence. This information can be added using the **Annotate with Flanking Sequence** tool:

Toolbox | **Add Information to Variants** ( ) | **Add Flanking Sequence** ( )

This opens a dialog where you can select a variant track ( ) to be annotated.

Clicking **Next** will display the dialog shown in figure 23.3

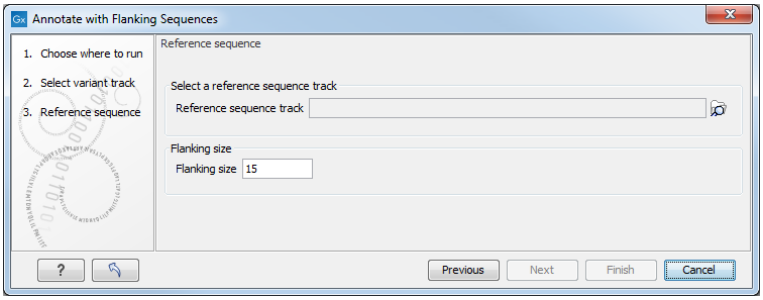


Figure 23.3: *Specifying a reference sequence and the amount of flanking bases to include.*

Select a sequence track that should be used for adding the flanking sequence, and specify how large the flanking region should be.

The result will be a new track with an additional column for the flanking sequence formatted like this: `CGGCT[T]AGTCC` with the base in square brackets being the variant allele.

23.5 Add fold changes

With this tool you can add the expression fold changes to your variants. You will create a copy of the input variant track and add the gene name and expression fold changes to this track. When

you have added the expression fold changes to the variant track, they can be seen in the tooltip when you zoom all the way in on the individual variants or in the table view.

You can create a fold change track with the tool **Create Fold Change Track** that is described in section 29.3.

To add fold changes, go to the toolbox:

Toolbox | **Add Information to Variants** ( ) | **Add Fold Changes** ( )

If you are connected to a server, you will first be asked where you would like to run the analysis. Next, you will be asked to select the variant track that you would like to add fold changes to (figure 23.4). To select the variant track, double-click on the file name or click once on the file and then on the arrow pointing to the right side in the middle of the wizard. Click on the button labeled **Next**.

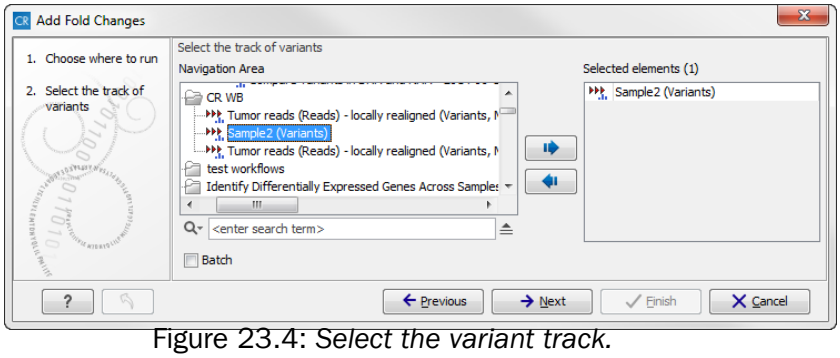


Figure 23.4: *Select the variant track.*

In the next step you can choose the fold change track (see figure 23.5).

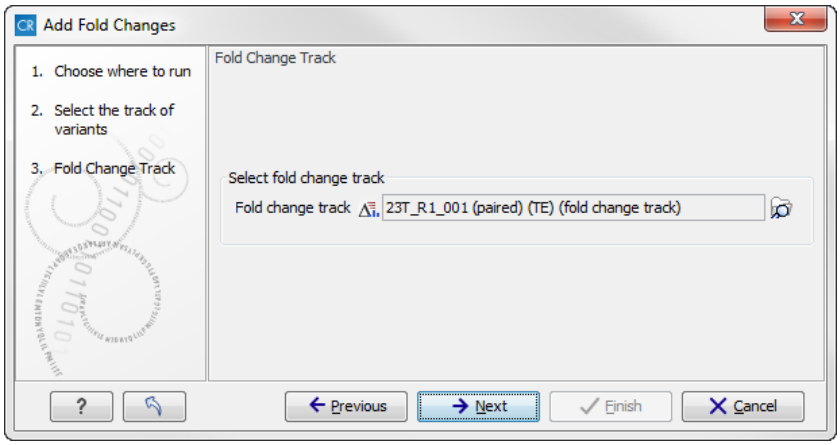


Figure 23.5: *Select the fold change track.*

Click on the button labeled **Next**, choose to save the results and click **Finish**.

The generated output is a variant track.If you open the variant track in table view by clicking on the table icon ( ) in the lower left corner of the **View Area**, you can see in the **Side Panel** under Table Settings that "Fold change" and "Gene" have been added to the list. If you would like to look into the numbers behind the fold changes, you can see the expression values in the original fold change file that was used as input in the "Add Fold Changes" analysis.

23.6 Add information about amino acid changes

This tool annotates variants with amino acid changes and creates a track for visual inspection of the amino acid changes. It takes a variant track as input and also requires a track with coding regions and a reference sequence.

To add information about amino acid changes to a variant track:

Toolbox | **Add Information to Variants** ( ) | **Add Information about Amino Acid Changes** ( )

If you are connected to a server, the first wizard step will ask you where you would like to run the analysis. Next, you must provide the variant track to be annotated with amino acid changes (see figure 23.6).

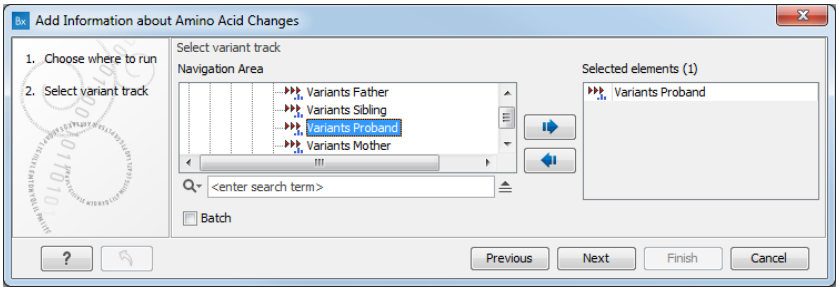


Figure 23.6: *The amino acid changes annotation tool takes variant tracks as input.*

Click on the button labeled **Next** to go to the next wizard step (figure 23.7).

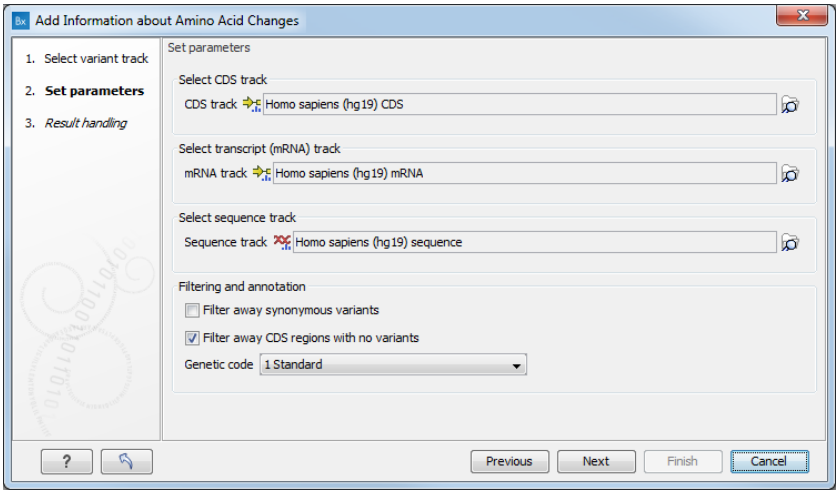


Figure 23.7: *Select CDS, mRNA, and sequence track and choose whether or not you would like to filter away synonymous variants.*

In this step you get the following options:

- **Select CDS track** Click on the folder icon ( ) in the right side of the wizard to select a CDS track. The CDS track is used to determine the reading frame and exon location to be used for translation. The CDS, mRNA, and sequence tracks can be found in the **Navigation Area** in the **CLC_References** folder.
- **Select mRNA track** Click on the folder icon ( ) in the right side of the wizard to select the mRNA track. The mRNA track is used to determine whether the variant is inside or outside

the region covered by the transcript. The mRNA track is optional. If you choose to provide an mRNA track you annotate variants that are located in the mRNA but also outside the region covering the coding sequence, in cases where such variants have been detected. If you choose not to provide an mRNA track, variants found outside the CDS region will not be annotated.

- **Select sequence track** Click on the folder icon () in the right side of the wizard to select the reference sequence track.
- **Filtering and annotation**
  - The "Filter synonymous variants" option allows you to filter away synonymous variants that does not cause any change to the encoded amino acid.
  - By default, the tool will filter out CDS regions that have no variants. You can choose to include them in your amino acid annotation track by deselecting the "Filter CDs regions with no variants" option.
  - The genetic code is the code that is used for amino acid translation (see http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi). The default option is "1 standard", the vertebrate standard code. If relevant, you can use the drop-down list to change to the genetic code that applies to you organism.

Click on the button labeled **Next**, choose whether you would like to open or save the results and click on the button labeled **Finish**.

Two types of outputs are generated:

1. **A variant track** that has been annotated with the amino acid changes. The added information can be accessed via the tooltips in the variant track or in the extra columns that have been added to the variant table. The extra columns provide information about the amino acid changes (see http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi). The variant track opens in track view and the table view can be accessed by clicking on the table icon found in the lower left corner of the **View Area**.
2. **An amino acid track** that displays a graphical presentation of the amino acid changes. The track is based on the CDS track and in addition to the amino acid sequence of the coding sequence, all amino acids that have been affected by variants are shown as individual amino acids below the amino acid track. Changes causing a frameshift are symbolized with two arrow heads, and variants causing premature stop are marked with an asterisk. An example is shown in figure 23.8.

For each variant in the input track, the following information is added:

- **Coding region change**. This describes the relative position on the coding DNA level, using the nomenclature proposed at http://varnomen.hgvs.org/. Variants inside exons and in the untranslated regions of the transcript will also be annotated with the distance to the nearest exon. E.g. "c.-4A>C" describes a SNV four bases upstream of the start codon, while "c.*4A>C" describes a SNV four bases downstream of the stop codon.
- **Amino acid change**. This describes the change on the protein level. For example, single amino-acid changes caused by SNVs are listed as "p.[Gly261Cys]", denoting that in the

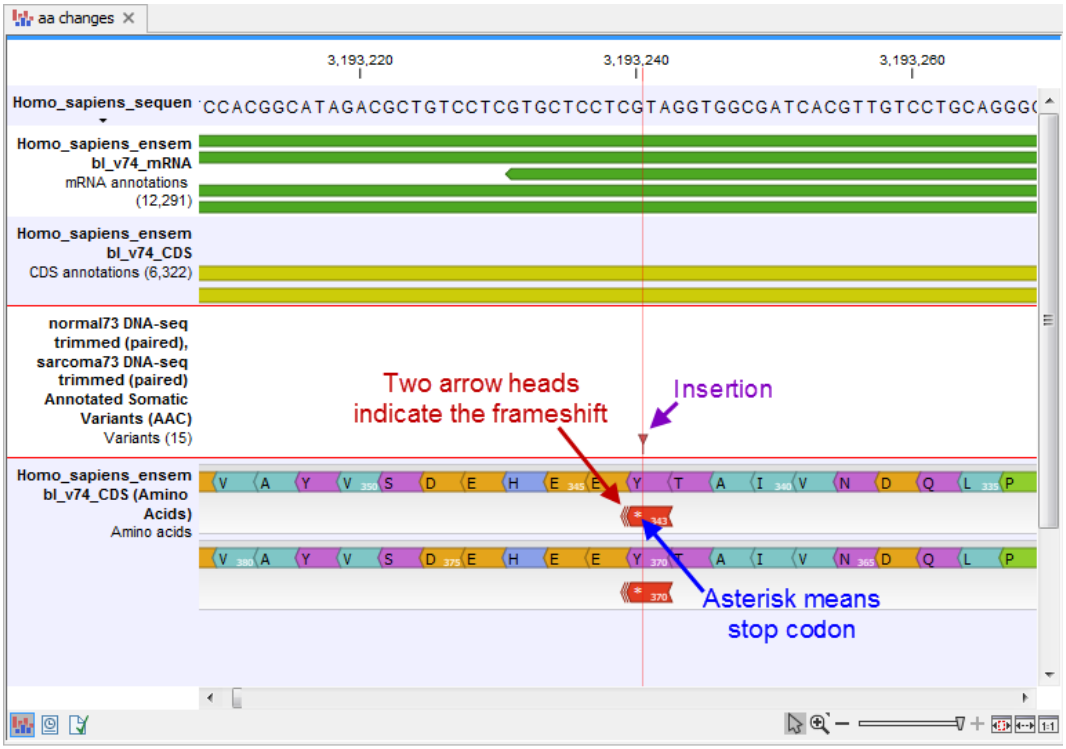


Figure 23.8: *The variant track and the amino acid track is here presented together with the reference sequence and the mRNA and CDS tracks. An insertion (purple arrow) has caused a frameshift (red arrow) that has changed an alanine to a stop codon (blue arrow).*

protein sequence (hence the "p.") the Glycine at position 261 is changed into Cysteine. Frame-shifts caused by nucleotide insertions and deletions are listed with the extension *fs*, for example p.[Pro244fs] denoting a frameshift at position 244 coding for Proline. For further details of the nomenclature see the "Recommendations for the description of protein sequence variants (v2.0)" at `http://varnomen.hgvs.org/`.

- **Coding region change in longest transcript**. When there are many transcript variants for a gene, the coding region change for all transcripts are listed in the "Coding region change" column. For quick reference, the longest transcript is often used, and there is a special column only listing the coding region change for the longest transcript.
- **Amino acid change in longest transcript**. This is similar to the above, just on the protein level.
- **Other variants within codon**. If there are other variants within the same codon, this column will have a "Yes". In this case, it should be manually investigated whether the two variants are linked by reads.
- **Non-synonymous**. Will have a "Yes" if the variant is non-synonymous at the protein level for any transcript. If the filter "Filter synonymous" was applied, this column will only contain entries labeled "Yes". A hyphen, "-", indicates the variant was present outside of a coding region.

An example of the output is given in figure 23.9.

Figure 23.9: *The resulting amino acid changes in track and table views. When the variant table has been opened by double-clicking on the text found in the left side of the View Area, the variant table and the variant track are linked. When clicking on an entry in the table, this position will be brought into focus in the variant track.*

The top track view displays a track list with the reference sequence, mRNA, CDS, variant, and amino acid tracks. The lower table view is the variant table that has been opened from the track list by double-clicking on the variant track name found in the left-hand side of the **View Area**. When opening the variant table in split view from the track list, the table and the variant track are linked.

An example illustrating a situation where different variants affect the same codon is shown in figure 23.10.

In this example three single nucleotide deletions are shown along with the resulting amino acid changes based on scenarios where only one deletion is present at the time. The first affected amino acid is shown for each of the three deletions. As the first deletion affect the encoded amino acid, this amino acid change is shown with a four nucleotide long arrow (that includes the deletion). The other two deletions do not affect the encoded amino acid as the frameshift was "synonymous" at the position encoded by the codon where the deletion was introduced. The

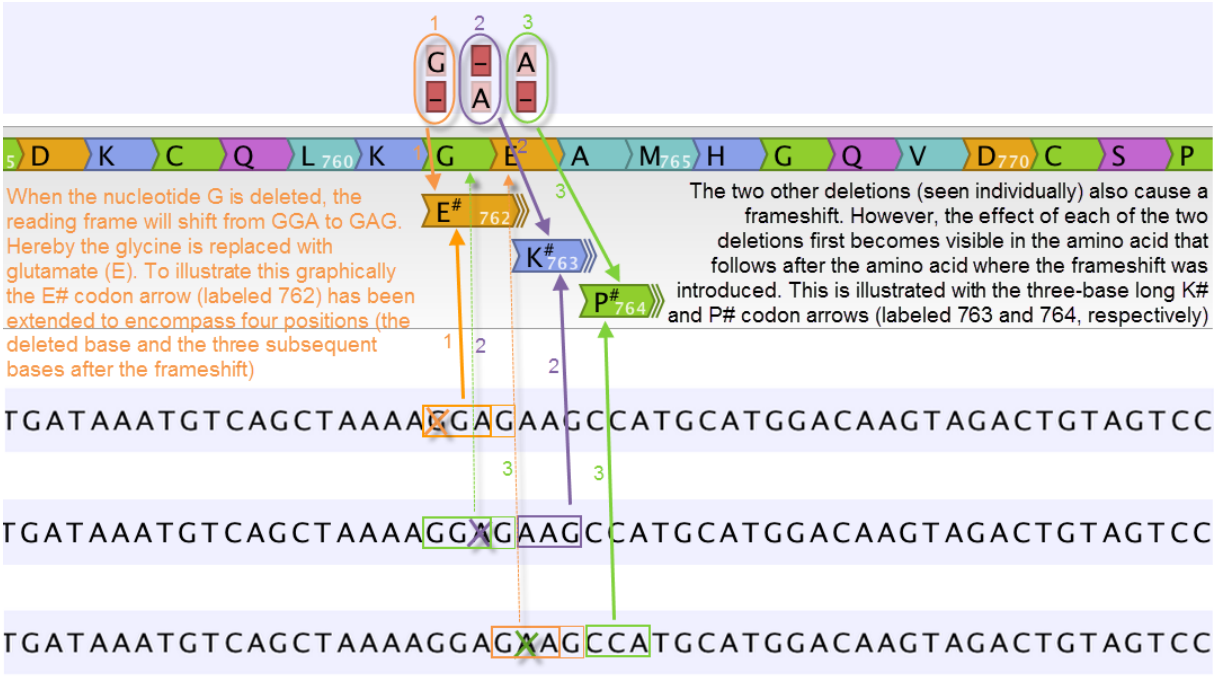


Figure 23.10: *Amino acids encoded from codons that potentially could have been affected by more than one variant are marked with a hash symbol (#) as the graphically presented amino acid changes always only include a single variant (a SNV, MNV, insertion, or deletion). Shown here are three different variants, present only one at the time, and the consequences of the three individual deletions. In cases where the deletion is found in a codon that is affected with an amino acid change, the arrow also include the deletion (situation 1) in the two other scenarios, the codon containing the deletion is changed to a codon that encodes the same amino acid, and the effect is therefore not seen until in the subsequent amino acid.*

effect is first seen at the next amino acid position (763 and 764, respectively), which does not contain a deletion, and therefore is illustrated with a three nucleotide long arrow.

The hash symbol (#) on the changed amino acids symbolize that more than one variant can be present in the region encoding this specific amino acid. The simultaneous presence of multiple variants within the same codon is not predicted by the amino acid changes tool. Manual inspection of the reads is required to be able to detect multiple variants within one codon.

The amino acid track

The colors of the amino acids in the amino acid track can be changed in the **Side Panel** under **Track layout** and "Amino acids track" (see figure 23.11).

Four different color schemes are available under "Amino acid colors":

- **Gray** All amino acids are shown in gray.
- **Group** Colors the amino acids in groups by the following properties:
  - **Purple** neutral, polar
  - **Turquoise** neutral, nonpolar
  - **Orange** acidic, polar
  - **Blue** basic ,polar

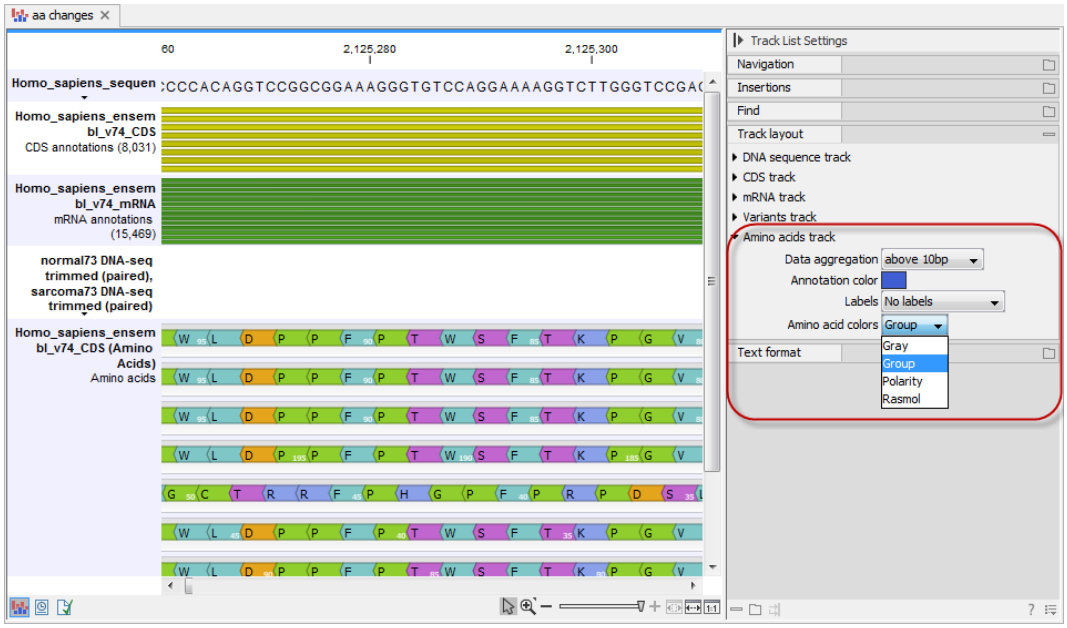


Figure 23.11: *The colors of the amino acids can be changed in the Side Panel under "Amino acids track".*

  - **Bright green** other (functional properties)

- **Polarity** Colors the amino acids according to the following categories:
  - **Green** neutral, polar
  - **Black** neutral, nonpolar
  - **Red** acidic, polar
  - **Blue** basic ,polar

- **Rasmol** Colors the amino acids according to the Rasmol color scheme (see `http://www.openrasmol.org/doc/rasmol.html`).

23.7 Add information from genomic regions

This will create a copy of the track used as input and add information from overlapping annotations or regions. To run the Add information from genomic regions tool, go to the toolbox:

> **Toolbox** | **Add Information to Variants** ( ) | **Add Information from genomic regions** ( )

First, select the track you wish to annotate and click **Next**. You can choose any kind of variant or annotation track as input. Next, select the track holding the overlapping region of interest (e.g. regulatory regions from ENCODE or if you have imported other databases containing regions that you would like to use for overlap comparison).

The result of this tool is a new track showing all the variants that now have been annotated with the information about the regions that overlap with the identified variants. The requirement for being registered as an overlap is that parts of the annotations are overlapping, regardless of the strandedness of the annotations. The added information can be visualized in two ways; 1) In the track tooltips when mousing over the individual variants or 2) in the table view where you can see

that new columns describing the added overlap information have been added to the table. The table view can be accessed by clicking on the table icon ( ) in the lower part of the **View Area**.

23.8 Add information from overlapping genes

This will create a copy of the track used as input and add information from overlapping genes and mRNA tracks as well as annotations from GeneCards. To run the Add information from overlapping genes tool, go to the toolbox:

Toolbox | **Add Information to Variants** ( ) | **Add Information from Overlapping Genes** ( )

First, select the track you wish to annotate and click **Next**. You can choose any kind of variant or annotation track as input. Next, select the a gene track and a mRNA track for overlap comparison (figure 23.12).

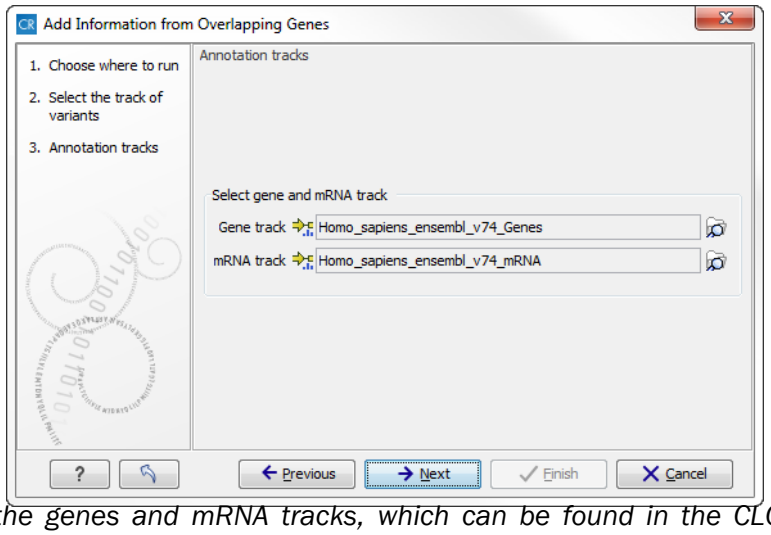


Figure 23.12: *Select the genes and mRNA tracks, which can be found in the CLC_References folder.*

You can find the gene and mRNA tracks in the CLC_References folder (figure 23.13).

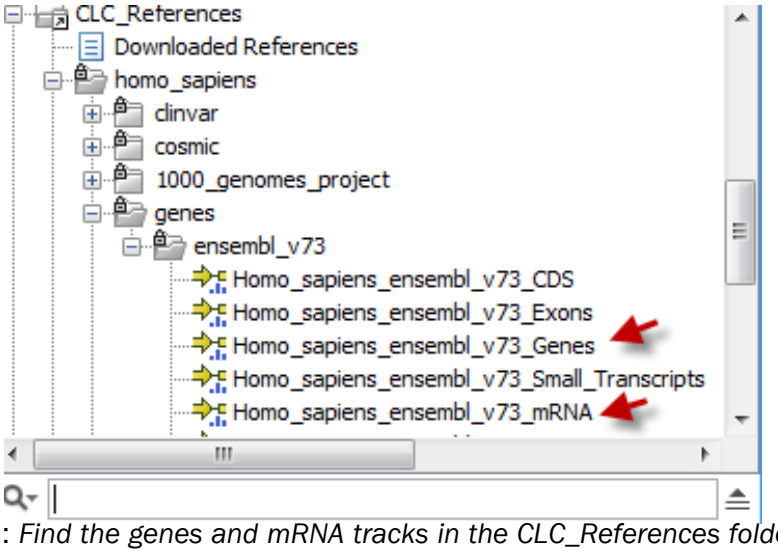


Figure 23.13: *Find the genes and mRNA tracks in the CLC_References folder.*

The result of this tool is a new track showing all the variants that now have been annotated with the information about genes and mRNA as well as annotations from GeneCards that overlap with the identified variants. The requirement for being registered as an overlap is that parts of the annotations are overlapping, regardless of the strandedness of the annotations (note that this makes it unsuitable for comparing e.g. two gene tracks but great for annotating variants with overlapping genes or regulatory regions). The added information can be visualized in two ways; 1) In the track tooltips when mousing over the individual variants or 2) in the table view where you can see that new columns describing the added gene and mRNA tracks have been added to the table. The table view can be accessed by clicking on the table icon () in the lower part of the **View Area**.

23.9 Link Variants to 3D Protein Structure

This tool makes it possible to visualize variant consequences on 3D protein structures. It takes a variant track as input, and produces a new variant track as output, with two additional columns in the table view:

- **Link to 3D protein structure:** If a variant affects the amino acid composition of a protein, and a 3D structure of sufficient homology can be found in the Protein Data Bank (PDB), a link is provided in this column. Via the link, the structure can be downloaded and a 3D model and visualization of the variant consequences on the protein structure will be created.
- **Effect on drug binding site:** If any of the homologous structures found in PDB has a drug or ligand in contact with the amino acid variation, a link is provided in this column. Via the link, a list of drug hits can be inspected. The list has links for creating 3D models and visualizations of the variant-drug interaction.

In section 23.8 it is described how to interpret the output in the variant table and how the tool finds appropriate protein structures to use for the visualizations, and in section 23.8 and onwards it is described how the 3D models and visualizations are created.

Note: Before running the tool, a protein structure sequence database must be downloaded and installed using the **Download 3D Protein Structure Database** tool (see section 23.10).

To run the tool, select:

Toolbox | **Add Information to Variants** () | **Link Variants to 3D Protein Structure** ()

If you are connected to a server, you will first be asked where you want to run the analysis. In the next wizard step you will be asked for an input file. The **Link Variants to 3D Protein Structure** accepts variant tracks as input (see figure 23.14).

Click on the button labeled **Next**. In the next wizard step, you must provide a CDS track and the reference sequence track (figure 23.15).

You can find the CDS and reference sequence in the **Navigation Area** under **CLC_References**.

Click on the button labeled **Next**, choose where you would like to save the data, and click on the button labeled **Finish**.

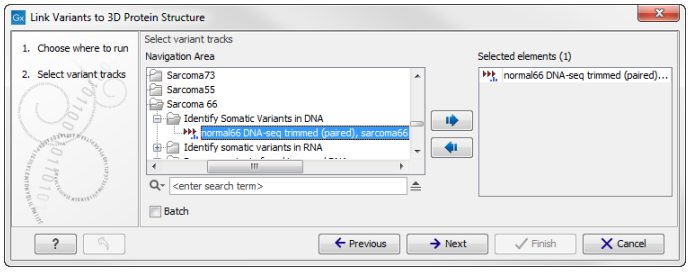


Figure 23.14: *Select the variant track holding the variants that you would like to visualize on 3D protein structures.*

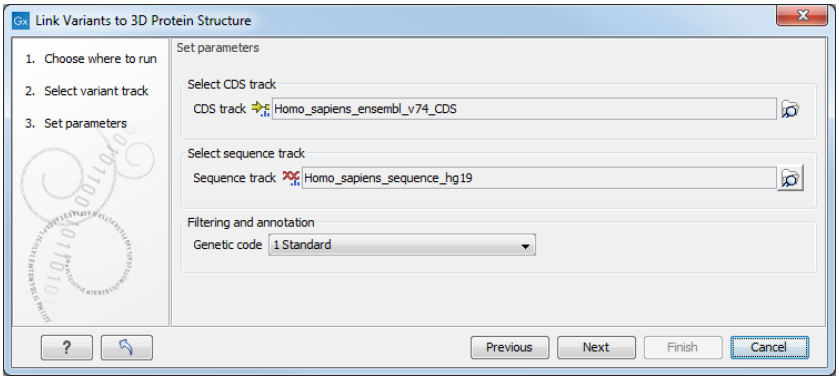


Figure 23.15: *Select CDS and reference sequence.*

As output, the tool produces a new variant track, with two additional columns in the table view ('Link to 3D protein structure' and 'Effect on drug binding site' - figure 23.16). The default output view is the variant track. To shift to table view, click on the table icon found in the lower left corner of the View Area.

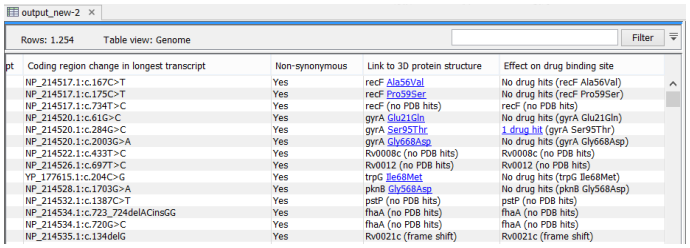


Figure 23.16: *The variant table output.*

The variant table output

For each variant in the input, the Link Variants to 3D Protein Structure tool does the following, to prepare output for the "Link to 3D protein structure" and "Effect on drug binding site" columns in the output variant track:

1. Evaluate if the variant is found inside a CDS region. Otherwise the following is returned for the variant: **(outside CDS regions)**.

2. If the variant is in a CDS region, translate the reference sequence of the impacted gene into an amino acid sequence and evaluate if the variant can be expected to have an effect on protein structure that can be visualized. Overlapping genes (common in prokaryotic

genomes) with different reading frames may cover a given variation, in which case multiple protein sequences will be considered.

For variants that cannot be visualized, the gene name and one of the reasons given below will be listed in the output table:

- **(nonsense)** - the variant would result in a stop codon being introduced in the protein sequence.
- **(synonymous)** - the variant would not change the amino acid.
- **(frame shift)** - the variant would introduce a frame shift.

3. BLAST the translated amino acid sequence (the query sequence) against the protein structure sequence database (see section 23.10) to identify structural candidates. Note that if multiple splicing variants exist, the protein structure search is based on the longest splicing variant. BLAST hits with E-value > 0.0001 are rejected and a maximum of 2500 BLAST hits are retrieved. If no hits are obtained, the gene name and the message **(no PDB hits)** are listed.

4. For each BLAST hit, check if the variant is covered by the structure. For a variant resulting in one amino acid being replaced by another, the affected amino acid position should be present on the structure. For a variant resulting in amino acid insertions or deletions, the amino acids on both sides of the insertion/deletion must be present on the structure.

5. For the BLAST hits covering the variant, rank the structures considering both structure quality and homology (see section 23.8).

6. Add the gene name and the description of the amino acid change to the "Link variant to 3D protein structure" column in the output variant track. A link on the description gives access to a 3D view of the variant effect using the best ranked protein structure from point 5 (see section 23.8). Note that the amino acid numbering is based on the longest CDS annotation found.

7. Extract all BLAST hits from point 5, where the affected amino acid(s) are in contact with a drug or ligand in the PDB file (heavy atoms within 5 Å). If no structures with variant-drug interaction are found, the following is returned to the "Effect on drug binding site" column: **No drug hits** together with the gene name and the description of the amino acid change. If structures with variant-drug interaction are found, the number of different drugs or ligands encountered are written to the "Effect on drug binding site" column as *X* **drug hits**. From a link on "*X* drug hits", a list describing the drug hits in more detail can be opened. The list also has a link for each drug, to create a 3D model and visualization of the variant-drug interaction section 23.8.

Create 3D visualization of variant

Clicking a link provided in the 'Link to 3D Protein Structure' column will show a menu with three options:

- 'Download and Show Structure' will open a 3D view visualizing the consequences of the variant on a protein structure (figure 23.16).

- 'Download and Show All Variants (x) on Structure' will open a 3D view visualizing the consequences of *x* variants on the same protein structure (figure 23.17).

  Note 1: Only variants shown in the table will be included in the view (e.g. variants filtered out will be ignored).

  Note 2: It is not always possible to visualize variants on the same gene together on the same structure, since many structures in the PDB only cover parts of the whole protein.

  Note 3: Even though variants may be possible to visualize together, it does not necessarily mean they occur together on the same protein. For example, in diploid cells, heterozygous variants may not.

- "Help" gives access to this documentation.

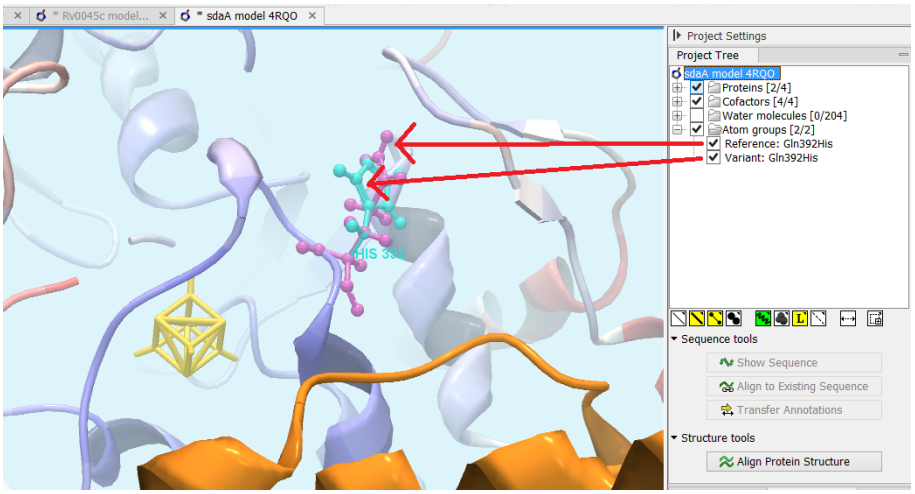


Figure 23.17: *Generated 3D view of variant. The reference amino acid is seen in purple and the variant in cyan on top of each other. Only the backbone of protein structures are visualized by default. The modeled protein structure is colored to indicate local model uncertainty - red for flexible and uncertain parts of the structure model and blue for very well defined and accurate parts of the structure model. Other molecules from the PDB file are colored orange or yellow.*

The "Download and Show.... " options will do the following:

1. **Download and import** the PDB file containing the protein structure for the variant (found by the 'Link Variants to 3D Protein Structure' tool - section 23.8).
2. **Generate biomolecule** involving the modeled chain, if the information is available in the PDB file (see Infobox below).
3. **Create an alignment** between the reference protein sequence for the gene impacted by the variant (the query sequence) and the sequence from the protein structure (the template structure).
4. **Create a model structure for the reference** by mapping it onto the template structure based on the sequence alignment (see section 23.8).
5. **Create a model structure with variant(s)** by mapping the protein sequence with the variant consequences onto the reference structure model (see section 23.8).

6. **Open a 3D view** (a Molecule Project) with the variant structure model shown in backbone representation. The model is colored by temperature (see figure 23.17), to indicate local model uncertainty (see section 23.8). The consequence(s) of the variant(s) are high-lighted by showing involved amino acids in ball n' sticks representation with the reference colored purple and the variant cyan. Other molecules from the PDB file are shown in orange or yellow coloring (figure 23.17).

From the Project Tree in the Side Panel of the Molecule Project, the category 'Atom groups' contains two entries for each variant shown on the structure - one entry for the reference and one for the variant (figure 23.17). The atom groups contain the visualization of the variant consequence on structure. For variants resulting in amino acid replacements, the affected amino acid is visualized. For variants resulting in amino acid insertions or deletions, the amino acids on each side of the deletion/insertion are visualized.

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).

Tip: Double-click an entry in the Project Tree to zoom the 3D view to the atoms.

You can save the 3D view (Molecule Project) in the Navigation Area for later inspection and analysis. Read more about how to customize visualization of molecules in section 11.3.

Infobox: Biomolecules in *Biomedical Genomics Workbench*

Protein structures imported from a PDB 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.
In *Biomedical Genomics Workbench* variants are therefore shown in a protein structure context representing a functioning biomolecule, if this information is available in the selected template PDB file.

Visualize drug interaction

Clicking a link provided in the 'Drug interaction in protein 3D structure' column will open a list with information about the drug hits (figure 23.18).

Rows: 7 Summary of hits found Filter

PDB hit	Drug name	DrugBank info	Organism	Description
2Y3P	SIMOCYCLINONE D8		ESCHERICHIA COLI	CRYSTAL STRUCTURE OF N-TERMINAL DOMAIN OF GYRA WITH THE ANTIBIOTIC SIMOCYCLINO
4KPF			STREPTOCOCCUS PNEUMONIAE	NOVEL FLUOROQUINOLONES IN COMPLEX WITH TOPOISOMERASE IV FROM S. PNEUMONIAE AN
3FOF	MOXIFLOXACIN (AVELOX)	DB00218	STREPTOCOCCUS PNEUMONIAE	STRUCTURAL INSIGHT INTO THE QUINOLONE-DNA CLEAVAGE COMPLEX OF TYPE IIA TOPOISON
3FOE	CLINAFLOXACIN		STREPTOCOCCUS PNEUMONIAE	STRUCTURAL INSIGHT INTO THE QUINOLONE-DNA CLEAVAGE COMPLEX OF TYPE IIA TOPOISON
3QX3	ETOPOSIDE (EPOSIN)	DB00773	HOMO SAPIENS	HUMAN TOPOISOMERASE IIBETA IN COMPLEX WITH DNA AND ETOPOSIDE
4G0V	MITOXANTRONE (MITOX)	DB01204	HOMO SAPIENS	HUMAN TOPOISOMERASE IIBETA IN COMPLEX WITH DNA AND MITOXANTRONE
4G0W	AMETANTRONE		HOMO SAPIENS	HUMAN TOPOISOMERASE IIBETA IN COMPLEX WITH DNA AND AMETANTRONE

Figure 23.18: *An example of a drug hit table with information about the drug and with links to 3D visualizations of variant-drug interaction.*

The information available in the drug hit list is the following:

- **PDB hit.** Clicking a link provided in the PDB hit column will show a menu with two options: "Download and Show Structure" and "Help". "Help" gives access to this documentation. The "Download and Show Structure" option does exactly as described in section 23.8, except that the final 3D visualization is centered on the drug, and the drug is shown in ball n' sticks representation with atoms colored according to their atom types (figure 23.19).
- **PDB drug name.** (Hidden by default) The identifier used by PDB for the ligand or drug.
- **Drug name.** When possible, a common name for the potential drug is listed here. The name is taken from the corresponding DrugBank entry (if available) or from the PDB header information for the PDB hit.
- **DrugBank info.** If information about the drug or ligand is available in DrugBank (www.drugbank.ca [Law et al., 2014, Wishart et al., 2006]), a weblink to the appropriate site is listed here.
- **E-value.** (Hidden by default) The E-value is a measure of the quality of the match returned from the BLAST search. The closer to zero, the more homologous is the template structure to the query sequence.
- **Organism.** (Hidden by default) The organism for which the PDB structure has been obtained.
- **Description.** The description of the PDB file content, as given in the header of the PBD hit.

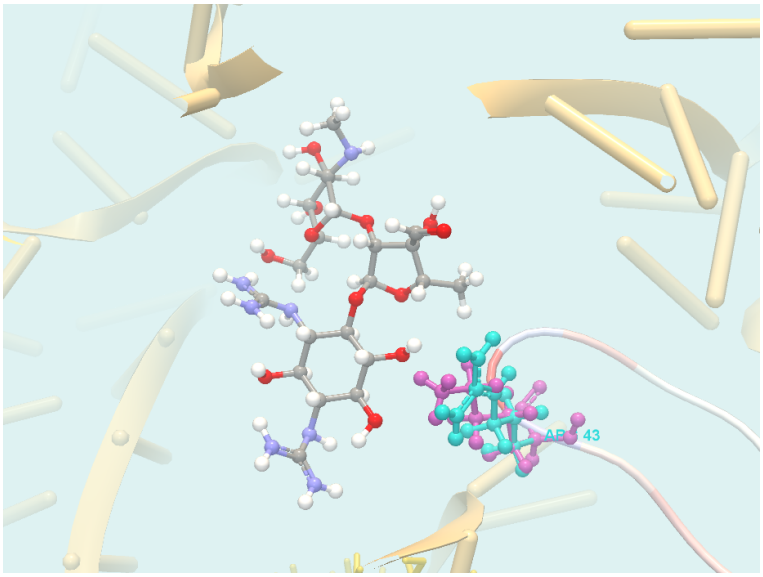


Figure 23.19: *3D visualization of variant-drug interaction. The drug (streptomycin) is in center of the view, and colored according to it's atom types. The variant is visualized as in figure 23.17. In this picture, the foreground has been cut away using the clipping plane functionality found in the Visualization tab in the Side Panel, to better see the variant-drug interaction.*

Protein coloring to visualize local structural uncertainties

The default coloring scheme for modeled structures in *Biomedical Genomics Workbench* is "Color by Temperature". This coloring indicates the uncertainty or disorder of each atom position in the structure.

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 $Å^2$, and is typically in the range [0, 100].

The temperature color scale ranges from blue (0) over white (50) to red (100) (see section 11.3.1).

For structure models created in *Biomedical Genomics 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 23.20 and 23.21.

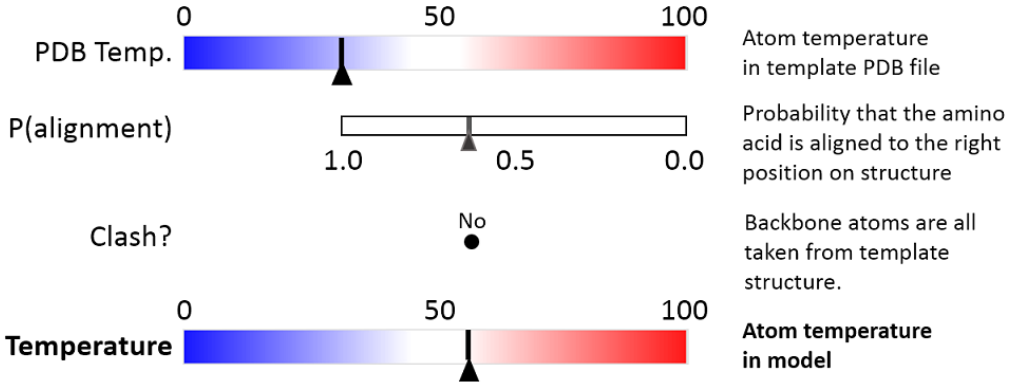


Figure 23.20: *Evaluation of temperature color for backbone atoms in structure models.*

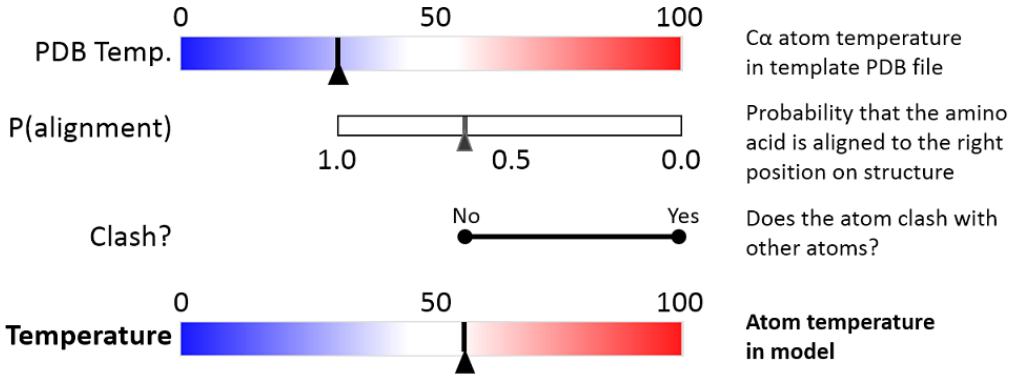


Figure 23.21: *Evaluation of temperature color for side chain atoms in structure models.*

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 23.22.

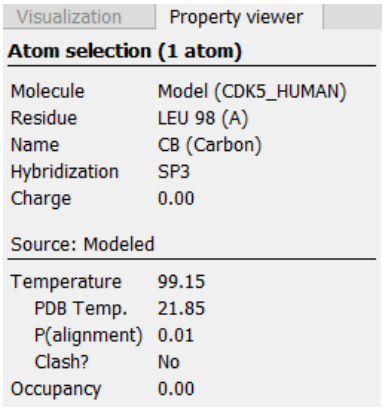


Figure 23.22: *Information displayed in the Side Panel Property viewer for a modeled atom.*

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(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:

```
Template  GGACDAEDRSTRSTACE---GG
Target    GGACD---RSTRSTACEKLMGG
```

is uncertain, because an alternative alignment is as likely:

```
Template  GGACDAEDRSTRSTACE---GG
Target    GGAC---DRSTRSTACEKLMGG
```

Clash?

Clashes are evaluated separately for each atom in a side chain. The scoring function used to evaluate protein-ligand interactions for docking and ligand optimization in the now discontinued *CLC Drug Discovery Workbench* is also used to evaluate the interaction between a given side chain atom and its surroundings.

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.

23.9.1 Method details

Ranking structures

The protein sequence of the gene affected by the variant (the query sequence) is BLASTed against the protein structure sequence database (section 23.10).

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 23.24. The five scaled descriptors are combined into the *template quality score*, weighting them to emphasize homology over structure qualities.

$$\text{Template quality score} = 3 \cdot S_{\text{E-value}} + 3 \cdot S_{\text{Identity}} + 1.5 \cdot S_{\text{Coverage}} + S_{\text{Resolution}} + 0.5 \cdot S_{\text{Rfree}}$$

E-value is a measure of the quality of the match returned from the BLAST search.

% Match identity is the identity between the query sequence and the BLAST hit in the matched region. It is evaluated as

$$\text{\% Match identity} = 100\% \cdot (\text{Identity in BLAST alignment})/L_{\text{B}}$$

where L_{B} is the length of the BLAST alignment of the matched region, as indicated in figure 23.23, 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 23.23). It is evaluated as

$$\text{\% Coverage} = 100\% \cdot (L_{\text{B}} - L_{\text{G}})/L_{\text{Q}}$$

where L_{G} is the total length of gaps in the BLAST alignment and L_{Q} is the length of the query sequence.

The **resolution** of a crystal structure is related to the size of structural features that can be resolved from the raw experimental data.

R_{free} is used to assess possible overmodeling of the experimental data.

Resolution and R_{free} are only given for crystal structures. NMR structures will therefore usually be ranked lower than crystal structures. Likewise, structures where R_{free} has not been given will tend to receive a lower rank. This often coincides with structures of older date.

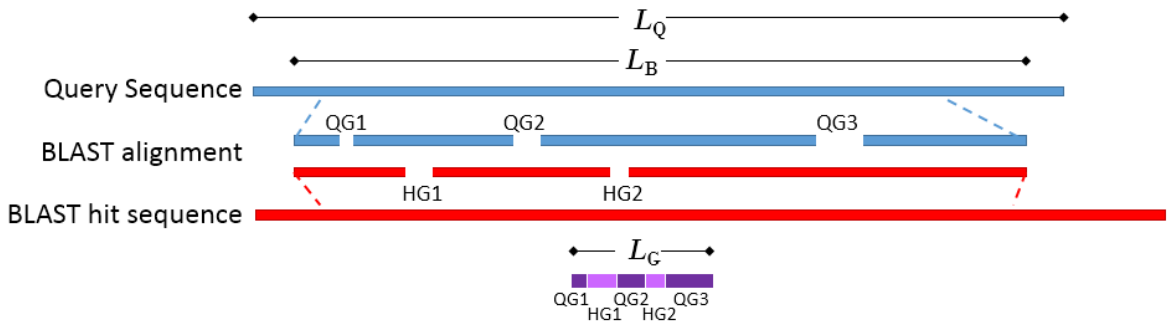


Figure 23.23: *Schematic of a query sequence matched to a BLAST hit. L_Q is the length of the query sequence, L_B is the length of the BLAST alignment of the matched region, QG1-3 are gaps in the matched region of the query sequence, HG1-2 are gaps in the matched region of the BLAST hit sequence, L_G is the total length of gaps in the BLAST alignment.*

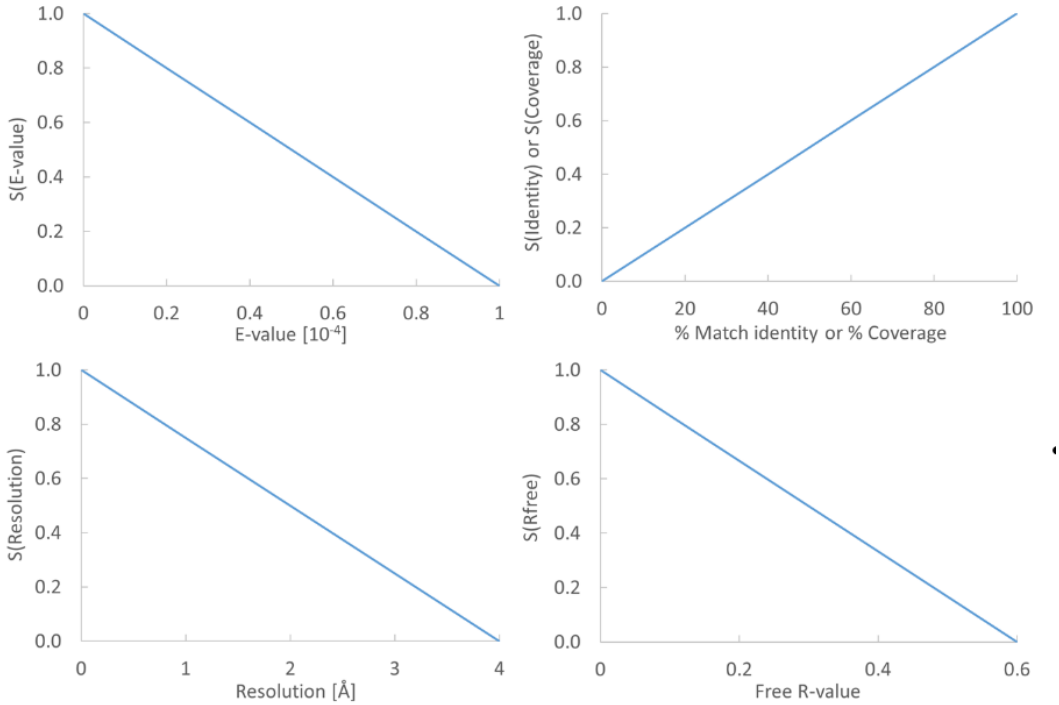


Figure 23.24: *From the E-value, % Match identity, % Coverage, Resolution, and Free R-value, the contributions to the "Template quality score" are determined from the linear functions shown in the graphs.*

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 23.25):

- For identical amino acids (example 1 in figure 23.25) => Copy atom positions from the PDB file. If the side chain is missing atoms in the PDB file, the side chain is rebuilt (section 23.8).
- For amino acid changes (example 2 in figure 23.25) => Copy backbone atom positions from the PDB file. Model side chain atom positions to match the query sequence (section 23.8).

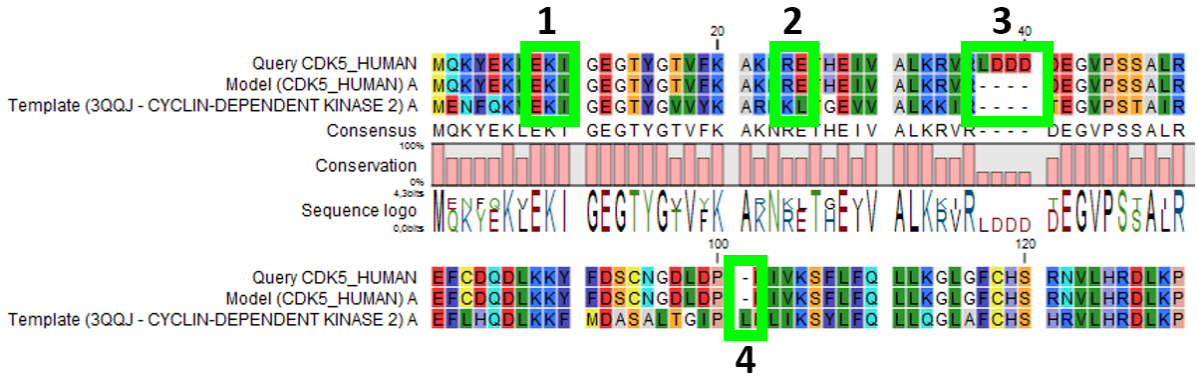


Figure 23.25: *Sequence alignment mapping query sequence (Query CDK5_HUMAN) to the structure with sequence "Template(3QQJ - CYCLIN-DEPENDENT KINASE 2)", producing a structure with sequence "Model(CDK5_HUMAN)". Examples are highlighted: 1. Identical amino acids, 2. Amino acid changes, 3. Amino acids in 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 amino acids in the query sequence not aligned to a position on the template structure (example 3 in figure 23.25) => No atoms are modeled. The model backbone will have a gap at this position and a "Structure modeling" issue is raised (see section 11.1).
- For amino acids on the template structure, not aligned to the query sequence (example 4 in figure 23.25) => The residues are deleted from the structure and a "Structure modeling" issue is raised (see section 11.1).

How side chains are modeled

Amino acid side chains tend to assume one of a discrete number of "rotamer" conformations. The rotamers used in *Biomedical Genomics 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 ϕ and ψ.
- 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.

23.10 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`) [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, select:

Toolbox | **Add Information to Variants** () | **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 23.26).

The downloaded database will be installed in the user home directory in a folder named CLCdatabases.

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).

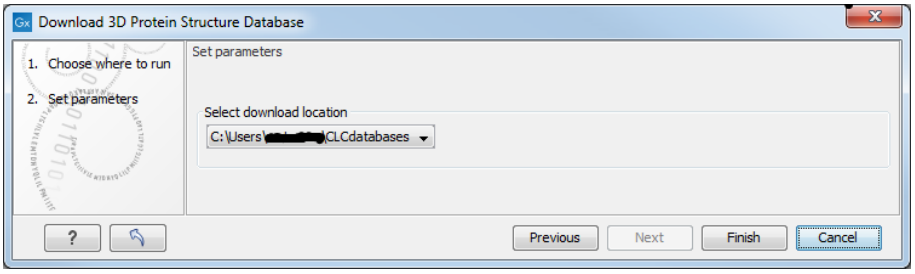


Figure 23.26: *Select the download location.*

23.11 From databases

23.11.1 Add information from 1000 Genomes Project

To run the Add information from 1000 Genomes Project tool, go to the toolbox:

> **Toolbox** | **Add Information to Variants** ( ) | **From Databases** ( )| **Add Information from 1000 Genomes Project** ( )

This tools adds information from variants identified by the 1000 Genomes Project to your variants. All you have to do when running this tool is to select the variant track that you want to annotate with information from the 1000 Genomes Project and specify where you would like to save the output.

23.11.2 Add information from COSMIC

To be able to run this tool, you must first download the COSMIC database directly from the Wellcome Trust Sanger Institute webpage and save it in the **Navigation Area**. Information about how to download COSMIC data can be found in section 6.2.

To run the Add information from COSMIC tool, go to the toolbox:

> **Toolbox** | **Add Information to Variants** ( ) | **From Databases** ( )| **Add Information from COSMIC** ( )

This tools adds information from known cancer variants in the COSMIC database to your variants. All you have to do when running this tool is to select the variant track that you want to annotate with information from the COSMIC database and select the COSMIC data that you have downloaded and saved in the Navigation Area.

23.11.3 Add information from ClinVar

To run the Add information from ClinVar tool, go to the toolbox:

> **Toolbox** | **Add Information to Variants** ( ) | **From Databases** ( )| **Add Information from Clinvar** ( )

This tools adds information from known clinically relevant variants in the ClinVar database to your variants.

23.11.4 Add information from common dbSNP

To run the Add information from common dbSNP tool, go to the toolbox:

Toolbox | **Add Information to Variants** () | **From Databases** ()| **Add Information from common dbSNP** ()

This tools adds information from common variants in the common dbSNP database to your variants.

23.11.5 Add information from HapMap

To run the Add information from HapMap tool, go to the toolbox:

Toolbox | **Add Information to Variants** () | **From Databases** ()| **Add Information from HapMap** ()

This tools adds information from common variants in the HapMap database to your variants.

23.11.6 Add information from dbSNP

To run the Add information from dbSNP tool, go to the toolbox:

Toolbox | **Add Information to Variants** () | **From Databases** ()| **Add Information from dbSNP** ()

This tools adds information from variants in the dbSNP database to your variants.

Chapter 24

Remove variants tools

Contents

Comparison with known variants from variant databases is a key concept when working with resequencing data. The *Biomedical Genomics Workbench* provides two types of tools for facilitating this task: one for *adding information to your experimental variants* with information from known variants (e.g. adding information about phenotypes like cancer associated with a certain variant allele), and one for *removing your experimental variants* based on this information (e.g. for removing common variants).

24.1 Remove variants found in external database

Any variant track can be used as "external database". It may either be produced by the *Biomedical Genomics Workbench*, imported or downloaded from variant database resources like dbSNP, 1000 genomes, HapMap etc. (see section 6.2). Hence, this tool has overlapping function with the three "From Databases" tools.

To run the Remove variants found in external database, go to the toolbox:

Toolbox | **Remove Variants** () | **Remove Variants Found in External Database** ()

(figure 24.1)

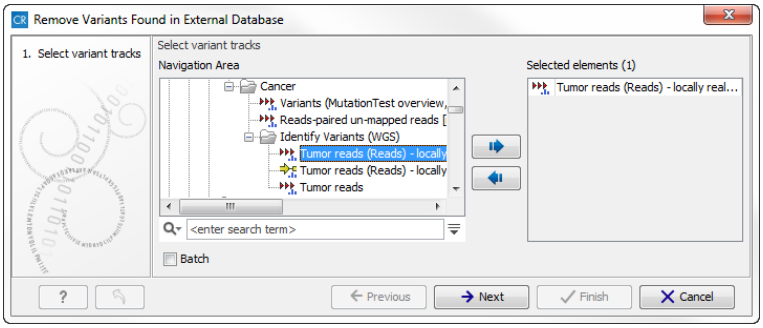


Figure 24.1: *Wizard Step 1.*

24.2 Remove variants not found in external database

To run the Remove variants not found in external database tool, go to the toolbox:

Toolbox | **Remove Variants** ( ) | **Remove Variants not Found in External Database** ( )

This tool removes variants that are found in your data set, but not in the specific external database, which you have imported into the *Biomedical Genomics Workbench* as a track. The track with the external variants has to be specified as a parameter to the tool.

24.3 Remove false positives

To run the Remove false positive, go to the toolbox:

Toolbox | **Remove Variants** ( ) | **Remove false positives** ( )

This tool will remove false positives by removing variants with low frequency, low average quality, and a bad forward/reverse balance.

After you have selected the variant track that you would like to remove false positives from, you can adjust the filter parameters to specify how many reads should be supporting a variant to pass the filter (figure 24.2).

The **Filter options** are:

- **Variant frequency** Checking this box allows to specify the minimum frequency %. Variants that are present below this frequency (calculated as 'count'/'coverage') will be removed.
- **Forward/reverse balance** Checking this box allows to specify the forward/reverse balance. Variants with a threshold below the specified threshold will be removed. E.g. if you check the forward/reverse balance setting and adjust this parameter to be a value greater than 0 and less than 0.5, it will then be necessary for at least two reads to be supporting the variant in order to pass this filter.
- **Average base quality** Checking this box allows to specify the minimum average base quality. Variants with an average base quality below the specified threshold will be removed.

24.4 Remove Germline Variants

Running the variant caller on a case and control sample separately and filtering away variants

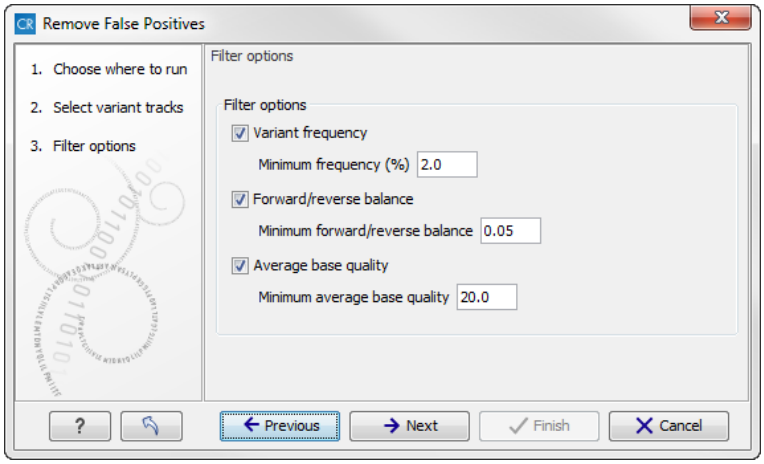


Figure 24.2: *This wizard step allows you to adjust the parameters for filtering.*

found in the control data set does not always give a satisfactory result as many variants in the control sample have not been called. This is often due to lack of read coverage in the corresponding regions or too stringent parameter settings. Therefore, instead of calling variants in the control sample, the Remove Germline Variants tool can be used to remove variants found in both samples from the set of candidate variants identified in the case sample.

Toolbox | **Remove Variants** ( ) | **Remove Germline Variants**

The variant track from the case sample must be used as input. When clicking **Next**, you are asked to supply the number of reads in the control data set that should support the variant allele in order to include it as a match (see figure 24.3). All the variants where at least this number of control reads show the particular allele will be filtered away in the result track.

Please note that variants, which have no coverage in the mapped control reads will be reported too. You can identify them by looking for a 0 value in the column 'Control coverage'.

The following annotations will be added to each variant not found in the control data set:

Control count For each allele the number of reads supporting the allele.

Control coverage Read coverage in the control dataset for the position in which the allele has been identified in the case dataset.

Control frequency Percentage of reads supporting the allele in the control sample.

The filter option can be used to set a threshold for which variants should be kept. In the dialog shown in figure 24.3 the threshold is set at two. This means that if a variant is found in one or less of the control reads, it will be kept.

24.5 Remove reference variants

The variant tracks produced by the variant detection tools of *Biomedical Genomics Workbench* include reference alleles complementing a non-reference allele (i.e. a heterozygous variant where only one allele is different from the reference). In some situations this information is not necessary and these reference allele variants can be filtered away. To run the Remove reference variants, go to the toolbox:

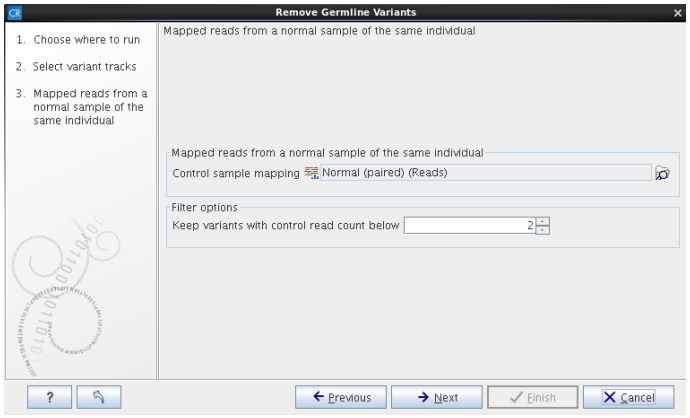


Figure 24.3: *Specify here the read mapping of the control sample and the minimum number of reads, which should include the variant before it will be removed as germline variant.*

Toolbox | **Remove Variants** ( ) | **Remove Reference Variants** ( )

This opens a wizard where you can select a variant track ( ) that should be filtered.

Click on the button labeled **Next** and **Finish** to create a new track without the reference variants.

24.6 Remove variants inside genome regions

To run the Remove variants inside genome regions, go to the toolbox:

Toolbox | **Remove Variants** ( ) | **Remove variants inside genome regions** ( )

This tool will remove variants that are present in specific genome regions. Genomic regions have to be available as a track and have to be specified as a parameter.

24.7 Remove variants outside genome regions

To run the Remove variants outside genome regions, go to the toolbox:

Toolbox | **Remove Variants** ( ) | **Remove variants outside genome regions** ( )

This tool will remove variants that are present outside specific genome regions. Genomic regions have to be available as a track and have to be specified as a parameter.

24.8 Remove variants outside targeted regions

The overlap filter will be used to filter an annotation track based on an overlap with another annotation track. This can be used to e.g. only show variants that fall within genes or regulatory regions or for restricting variant results to only cover a subset of genes as explained in section 28.4. Please note that for comparing variant tracks, more specific filters should be used (see chapter 23).

To run the Remove variants outside targeted regions, go to the toolbox:

Toolbox | **Remove Variants** ( ) | **Remove variants outside targeted regions** ( )

Select the track you wish to filter and click on the button labeled **Next** to specify the track of

overlapping annotations (see figure 24.4).

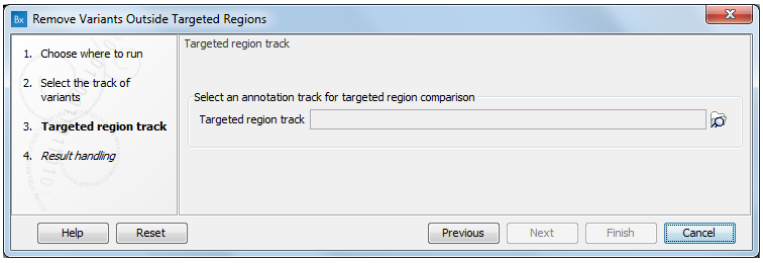


Figure 24.4: *Select overlapping annotations track.*

Next, select the track that should be used for comparison and tick whether you wish to keep annotations that overlap, or whether to keep annotations that do not overlap with the selected track. An overlap has a simple definition – if the annotation used as input has at least one shared position with the other track, there is an overlap. The boundaries of the annotations do not need to match.

24.9 From databases

24.9.1 Remove variants found in 1000 genomes project

To run the Remove variants found in 1000 genomes project, go to the toolbox:

> **Toolbox** | **Remove Variants** ( ) | **From Databases** ( )| **Remove Variants Found in 1000 Genomes Project** ( )

This tool will remove variants present in 1000 Genomes Data.

24.9.2 Remove variants found in common dbSNP

To run the Remove variants found in common dbSNP, go to the toolbox:

> **Toolbox** | **Remove Variants** ( ) | **From Databases** ( )| **Remove Variants Found in common dbSNP** ( )

This tool will remove variants present in the common dbSNP database with common variants in a specific population (population frequency > 1%).

24.9.3 Remove variants found in HapMap

To run the Remove variants found in HapMap, go to the toolbox:

> **Toolbox** | **Remove Variants** ( ) | **From Databases** ( )| **Remove variants found in HapMap** ( )

This tool will remove variants present in the HapMap database with common variants in a specific population.

Chapter 25

Add information to genes tool

Contents

25.1 Add information from overlapping variants

This will create a copy of the track used as input and add information from overlapping annotations or variants:

The result of this tool is a new track with all the annotations from the input track and with additional information from the annotations that overlap from the other track. Annotations are visible in the tooltips that appears when hovering the mouse on a variant in the Track view, or as additional columns in the Table view of the track.

Chapter 26

Compare samples tools

Contents

26.1 Compare shared variants within a group of samples

This tool should be used if you are interested in finding common (frequent) variants in a group of samples. For example one use case could be that you have 50 unrelated patients with the same disease and would like to identify variants that are present in at least 70% of all patients. It can also be used to do an overall comparison between samples (a frequency threshold of 0% will report all alleles).

> **Toolbox** | **Compare Samples** ( ) | **Compare Shared Variants within a Group of Samples** ( )

This opens a dialog where you can select the variant tracks ( ) from the samples in the group.

Clicking **Next** will display the dialog shown in figure 26.1.

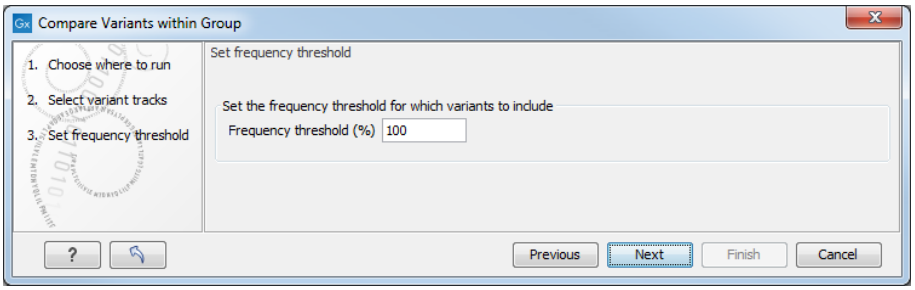


Figure 26.1: *Frequency treshold.*

The **Frequency threshold** is the percentage of samples that have this variant. Setting it to 70% means that at least 70% of the samples selected as input have to contain a given variant for it to be reported in the output.

The output of the analysis is a track with all the variants that passed the frequency thresholds and with additional reporting of:

Sample count The number of samples that have the variant

Total number of samples The total number of samples (this will be identical for all variants).

Sample frequency This is the same frequency that is also used as a threshold (see figure 26.1).

Origin tracks A comma-separated list of the name of the tracks that contain the variant.

Note that this tool can be used for merging all variants from a number of variant tracks into one track by setting the frequency threshold to 0.

26.2 Identify Enriched Variants in Case vs Control Group

This tool should be used if you have a case-control study. This could be patients with a disease (case) and healthy individuals (control). The idea is to identify variants which are more common in the case samples than in the control samples.

The Fisher exact test is applied on the number of occurrences of each allele of each variant in the case and the control data set. The alleles from each variant are considered separately, i.e. for an SNV with two alleles; a Fisher Exact test will be applied to each of the two. The test will also check whether an SNV in the case group is part of an MNV in the control group. Those with a low p-value are potential candidates for variants playing a role in the disease/phenotype. Please note that a low p-value can only be reached if the number of samples in the data set is high.

The tool is found in the Toolbox:

> **Toolbox** | **Compare Samples** () | **Identify Enriched Variants in Case vs Control Group** ()

In the first step of the dialog, you select the case variant tracks. Clicking **Next** shows the dialog in figure 26.2.

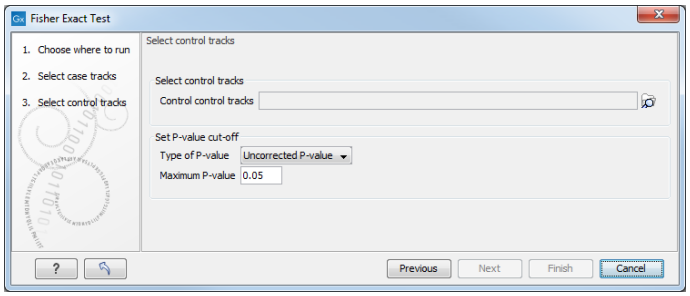


Figure 26.2: *In this dialog you can select the control tracks, a p-value correction method, and specify the p-value threshold for the fisher exact test.*

At the top, select the variant tracks from the control group. Furthermore, you must set a threshold for the p-value (default is 0.05); only variants having a p-value below this threshold will be reported. You can choose whether the threshold p-value refers to a corrected value for multiple tests (either Bonferroni Correction, or False Discovery Rate (FDR)), or an uncorrected p-value. A variant table is created as output (see figure 26.3), reporting only those variants with p-values

lower than the threshold. All corrected and uncorrected p-values are shown here, so alternatively, variants with non-significant p-values can also be filtered out or more stringent thresholds can be applied at this stage, using the manual filtering options.

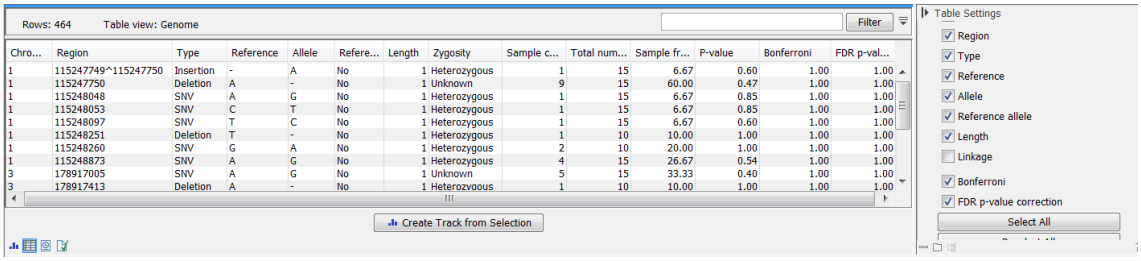


Figure 26.3: *In the output table, you can view information about all significant variants, select which columns to view, and filter manually on certain criteria.*

There are many other columns displaying information about the variants in the output table, such as the type, sequence, and length of the variant, its frequency and read count in case and control samples, and its overall zygosity. The zygosity information refers to **all** of the case samples; a label of 'homozygous' means the variant is homozygous in all case samples, a label of 'heterozygous' means the variant is heterozygous in all case samples, whereas a label of 'unknown' means it is heterozygous in some, and homozygous in others.

Overlapping variants: If two different types of variants occur in the same location, these are reported separately in the output table. This is particularly important, where SNPs occur in the same position as an MNV. Usually, multiple SNVs occurring alongside each other would simply be reported as one MNV, but if one SNV of the MNV is found in additional case samples by itself, it will be reported separately. For example, if an MNV of AAT -> GCA at position 1 occurs in five of the case samples, and the SNV at position 1 of A -> G, occurs in an *additional* 3 samples (so 8 samples in total), the output table will list the MNV and SNV information separately (however, the SNV will be shown as being present in only 3 samples, as this is the number in which it appears 'alone').

The test will also check whether an SNV in the case group is part of an MNV in the control group.

26.3 Trio analysis

This tool should be used if you have a trio study with one child and its parents. It should be mainly used for investigating differences in the child in comparison to its parents.

To start the Trio analysis:

Toolbox | **Compare Samples** ( ) | **Trio Analysis** ( )

In the first step of the dialog, select the variant track of the child. Clicking **Next** shows the dialog in figure 26.4.

Click on the folder ( ) to select the two variant tracks for the mother and the father. In case you have a human TRIO, please specify if the child is male or female and how the X, Y chromosomes as well as the mitochondrion are named in the genome track. These parameters are important in order to apply specific inheritance rules to these chromosomes.

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

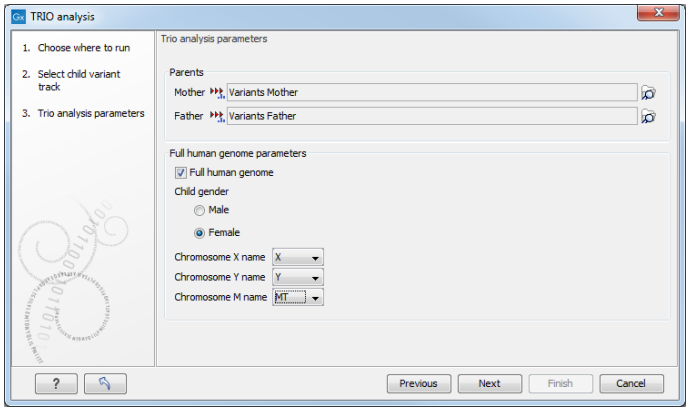


Figure 26.4: *Selecting variant tracks of the parents.*

The output is a variant track showing all variants detected in the child. For each variant in the child, it is reported whether the variant is inherited from the father, mother, both, either or is a de novo mutation. This information can be found in the tooltip for each variant or by switching to the table view (see the column labeled "Inheritance") (figure 26.5).

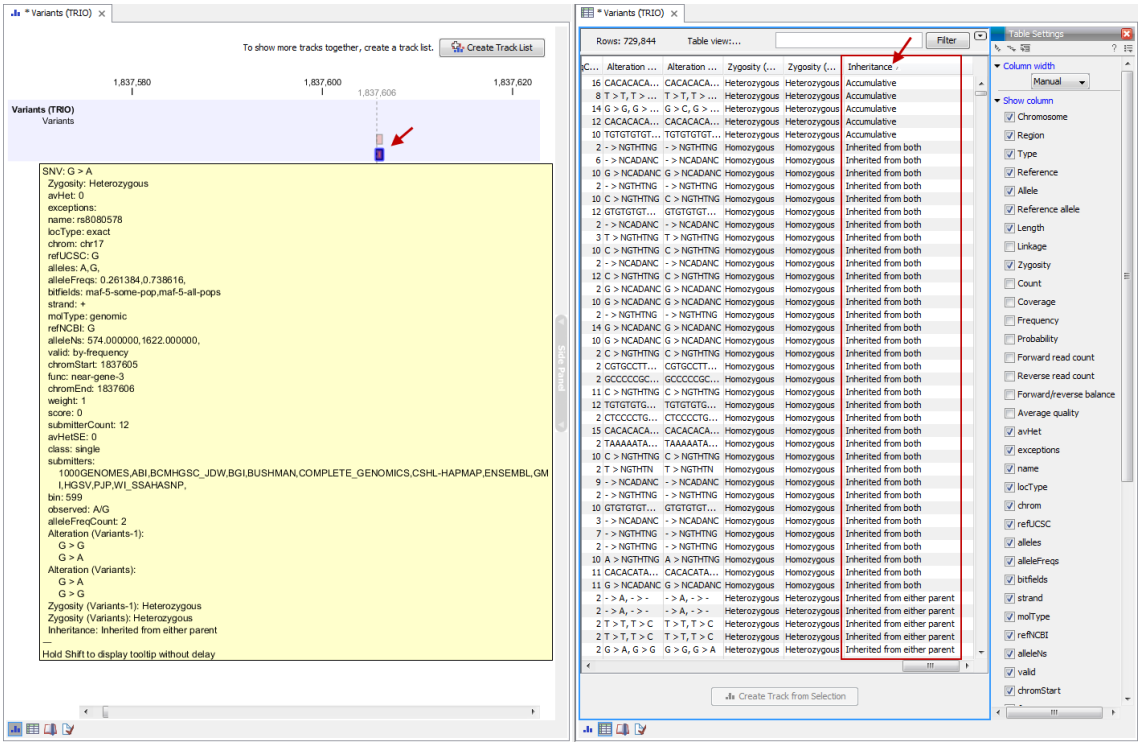


Figure 26.5: *Output from Trio Analysis showing the variants found in the child in track and table format.*

In cases where both parents are heterozygous with respect to a variant allele, and the child has the same phenotype as the parents, it is unclear which allele was inherited from which parent. Such mutations are described as 'Inherited from either parent'.

In cases where both parents are homozygous with respect to a variant allele, and the child has the same phenotype as the parents, it is also unclear which allele was inherited from which parent. Such mutations are described as 'Inherited from both parents'.

In cases where both parents are heterozygous and the child homozygous for the variant, the child

has inherited a variant from both parents. In such cases the tool will also check for a potential recessive mutation. Recessive mutations are present in a heterozygous state in each of the parents, but are homozygous in the child. To investigate potential disease relevant variants, recessive variants and de novo variants are the most interesting (in case the parents are not affected). The tool will also add information about the genotype (homozygote or heterozygote) in all samples.

For humans, special rules apply for chromosome X (in male children) and chromosome Y, as well as the mitochondrion, as these are haploid and always inherited from the same parent. Heterozygous variants in the child that do not follow mendelian inheritance patterns will be marked in the result.

Let's look at an example where these special rules apply - in this case the trio analysis is performed with a boy:

The boy has a position on the Y chromosome that is heterozygous for C/T. The heterozygous C is not present in neither the mother or father, but the T is present in the father. In this case the inheritance result for the T variant will be: 'Inherited from the father', and for the C variant 'de novo'. However, both variants will also be marked with 'Yes' in the column 'Mendelian inheritance problem' because of this aberrant situation. In case the child is female, all variants on the Y chromosome will be marked in the same way.

The following annotations will be added to the resulting child track:

Zygosity Zygosity in the child as reported from the variant caller. Can be either homozygote or heterozygote.

Zygosity (Name of parent track 1) Zygosity in the corresponding parent (e.g. father) as reported from the variant caller. Can be either homozygote or heterozygote.

Allele variant (Name of parent track 1) Alleles called in the corresponding parent (e.g. father).

Zygosity (Name of parent track 2) Zygosity in the corresponding parent (e.g. mother) as reported from the variant caller. Can be either homozygote or heterozygote.

Allele variant (Name of parent track 2) Alleles called in the corresponding parent (e.g. mother).

Inheritance Inheritance status. Can be one of the following values, or a combination of these: 'De novo', 'Recessive', 'Inherited from both', 'Inherited from either', 'Inherited from (Name of parent track)'. For example, a proband homozygous for a variant that only one parent (the mother in this case) presents will result in an inheritance described as "De novo, inherited from mother".

Mendelian inheritance problem Variants not following the mendelian inheritance pattern are marked here with 'Yes'.

Note! If the variant at this position cannot be found in either of the parents, the zygosity status of the parent where the variant has not been found is unknown, and the allele variant column will be left empty.

Chapter 27

Identify candidate variants tools

Contents

27.1 Identify candidate variants

After the variants have been identified and post-filtered (e.g. for somatic variants), the next task is to identify e.g. driver mutations or to identify those variants that should be validated first.

The **Identify Candidate Variants** tool can be used to identify and extract variants that fulfill certain criteria. This is done using filter criteria that is defined in a wizard step, while running the **Identify Candidate Variants** tool. The **Identify Candidate Variants** tool takes tracks with annotated or non-annotated variants as input. To run the Identify candidate variants, go to the toolbox:

Toolbox | **Identify Candidate Variants** () | **Identify candidate variants** ()

An example of a filter criterion is shown in figure 27.1 where you filter away all variants that are not found on chromosome 1 and that are not homozygous. As a result you will only keep the variants that are homozygous and found on chromosome 1.

The best way to use the tool is to use your variant track as a guidance variant track to load all the column headers into a filter criteria drop-down list (see figure 27.1). This is done in the following way:

1. Click on the folder icon in the wizard.
2. Select a variant track that has the terms that you would like to filter on (this variant track is only used as a guidance track that allows you to choose the column headers (or other words that can be used for filtering) from a drop down list. The alternative is to write the column headers manually).
3. Click on the button labeled "Load Annotations" to load your annotated variant track that is to be used as template. The track is used as a guidance variant track in order to obtain

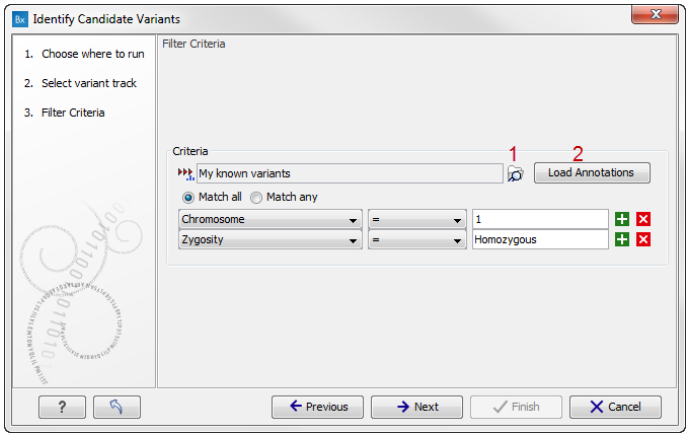


Figure 27.1: *Create filter criteria to extract the variants of interest. In this example we will extract the homozygous variants found on chromosome 1. Click on the folder icon (marked with the number 1) to select a guidance track that allows you to easily select the available column headers that can be used for filtering. Click on "Load Annotations" (marked with the number 2) to load the annotations.*

the annotation categories (annotation column headers) to be used for selection of which annotations the filter should be applied to.

Caution! Please note, that when you create filter criteria, you should use a guiding variant track that contains the same annotations as the annotations that are present in the variant tracks that you would like to filter.

The output from the **Identify Candidate Variants** tool is a variant track (and table) that contains only the variants that fulfill the specified filter criteria.

27.2 Remove information from variants

When you use information from various databases to annotate your variants, you may end up with many duplicated annotations or even annotations that you are not really interested in. This tool can help you remove annotations that have been added to variants, so that the results track/results table only includes the information that is of relevance to you.

To run the Remove information from variants, go to the toolbox:

Toolbox | **Identify Candidate Variants** ( ) | **Remove information from variants** ( )

The input for the tool is an annotated variant track (please make sure that you select a variant track that contains annotations e.g. Amino Acid Change, Exact Match, Conservation Score etc.). If you click on the button labeled "Load Annotations" in the wizard shown in figure 27.2 , the annotations that have been added to the variants in the input track are preloaded in the window below. You can choose, which annotations should be kept or removed. Please use the Ctrl or Shift keys on your keyboard to select the annotations.

However, when the tool **Remove information from variants** is used in a workflow, the option "Load Annotations" is not available and you will have to enter manually the annotation names you want to remove (or that you want to keep, depending on the choice made below in the "to keep or remove" drop-down menu).

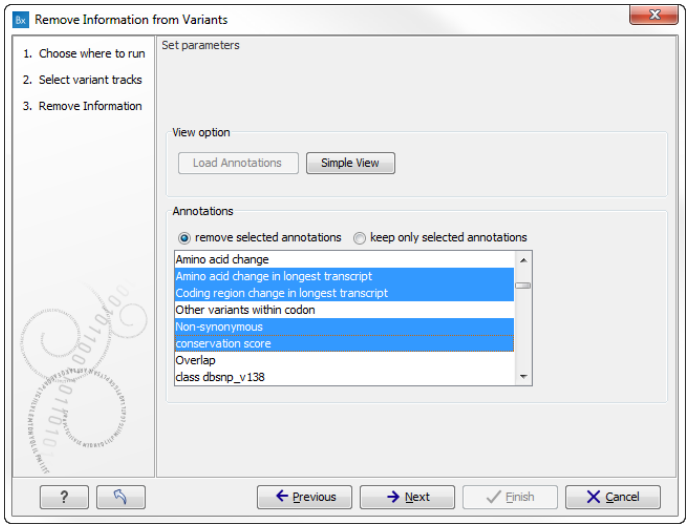


Figure 27.2: *The selected annotation columns preload from the input variant track will be removed in the output variant track.*

To know (and potentially copy) the names of the annotations you need to enter in the window of the workflow wizard, you can start the **Remove information from variants** tool on its own using as input a similar variant track than the one you will generate with the workflow. Select in the "Load Annotations" menu the annotation names you are interested in removing/keeping. Once you have completed your selection you can click on the button "Simple View" where only the selected annotations will be shown. You can now copy these annotation names, cancel the wizard of the stand alone tool and start the workflow wizard. Paste your selection in the relevant window after having started the workflow wizard.

27.3 Identify variants with effect on splicing

This tool will analyze a variant track to determine whether the variants fall within potential splice sites. First select your variant track (figure 27.3) followed by a transcript track (see figure 27.4). As part of the dialog you can choose to exclude all variants that do not fall within a splice site.

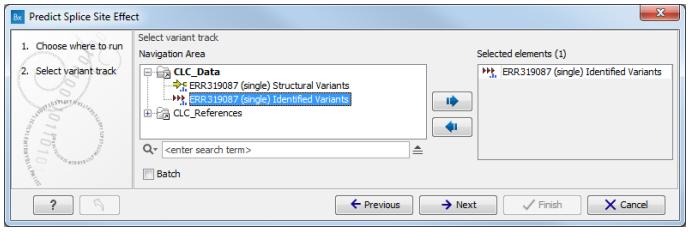


Figure 27.3: *Variant track selection.*

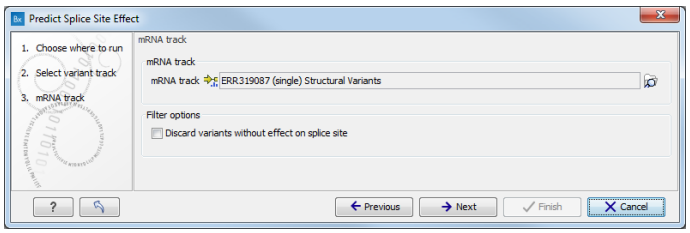


Figure 27.4: *Transcript track selection.*

If a variant falls within two base pairs of an intron-exon boundary, it will be annotated as a possible splice site disruption (see figure 27.5).

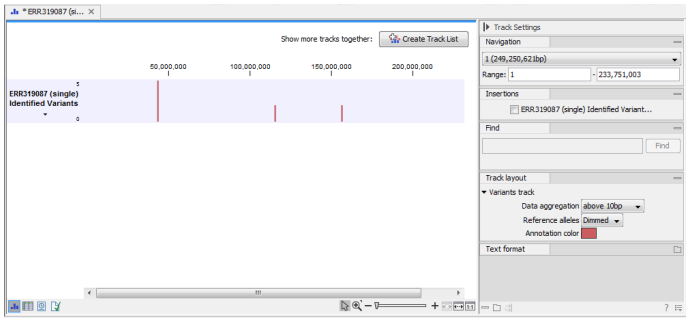


Figure 27.5: *Results of the predict splice site effect tool.*

Chapter 28

Identify candidate genes tools

Contents

28.1 Identify differentially expressed gene groups and pathways

This tool can be used to investigate candidate differentially expressed genes for a common functional role. For example if you would like to compare different cancer patients to check whether e.g. the same pathways are affected in different individuals, you can use this tool.

For this, you need a GO association file, which includes gene names and associated Gene Ontology terms. A GO association file with the top-level GO terms annotated (GO slim) is provided with the *Biomedical Genomics Workbench* and can be downloaded using the **Data Management** () function found in the top right corner of the Workbench (see section 13.1).

To run the analysis go to the toolbox:

> **Toolbox** | **Identify Candidate Genes** () | **Identify Differentially Expressed Gene Groups and Pathways** ()

When you run the Identify Differentially Expressed Gene Groups and Pathways analysis, you first have to select the expression comparison track () you wish to annotate with the GO term enrichment analysis. Expression comparison tracks can be created e.g. by the create fold change track tool (see section 29.3).

After clicking **Next**, you have to specify the annotation association file, a gene track, and finally which ontology (cellular component, biological process or molecular function) you would like to test for (see figure 28.1).

Next, the Workbench tries to match gene names from the expression comparison track with the gene names in the GO association file. Please be aware that the same gene name definition should be used in both files.

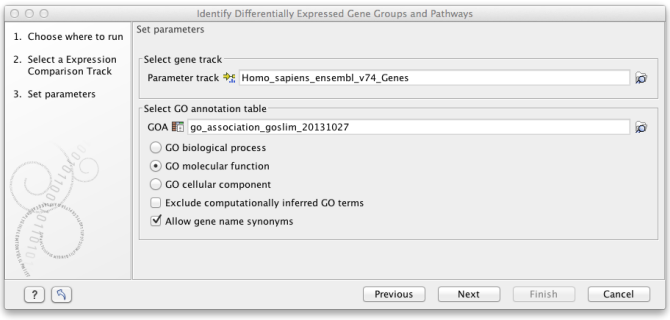
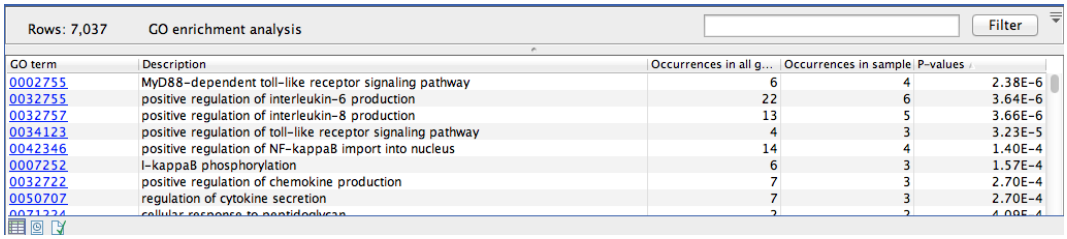


Figure 28.1: *Select gene track, GO annotation table, and ontology.*

Based on this, the Workbench finds GO terms that are over-represented in the list. A hypergeometric test is used to identify over-represented GO terms by testing whether some of the GO terms are over-represented in a given gene set, compared to a randomly selected set of genes.

The result is a table with GO terms and the calculated p-value for the differentially expressed genes, and a new expression comparison track with annotated GO terms and the corresponding p-value (see figure 28.2). The p-value is the probability of obtaining a test statistic at least as extreme as the one that was actually observed, or in other words how significant (trustworthy) a result is. In case of a small p-value the probability of achieving the same result by chance with the same test statistic is very small.



GO term	Description	Occurrences in all g...	Occurrences in sample	P-values
0002755	MyD88-dependent toll-like receptor signaling pathway	6	4	2.38E-6
0032755	positive regulation of interleukin-6 production	22	6	3.64E-6
0032757	positive regulation of interleukin-8 production	13	5	3.66E-6
0034123	positive regulation of toll-like receptor signaling pathway	4	3	3.23E-5
0042346	positive regulation of NF-kappaB import into nucleus	14	4	1.40E-4
0007252	I-kappaB phosphorylation	6	3	1.57E-4
0032722	positive regulation of chemokine production	7	3	2.70E-4
0050707	regulation of cytokine secretion	7	3	2.70E-4

Figure 28.2: *The results of the analysis.*

28.2 Identify highly mutated gene groups and pathways

This tool can be used to investigate candidate variants or better their corresponding altered genes for a common functional role. For example if you would like to compare different cancer patients to check whether e.g. the same pathways are affected in different individuals, you can use this tool. For this, you need a GO association file, which includes gene names and associated Gene Ontology terms. A GO association file with the top-level GO terms annotated (GO slim) is provided with the Biomedical Genomics Workbench and can be downloaded using the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

To run the analysis go to the toolbox:

> **Toolbox** | **Identify Candidate Genes** ( ) | **Identify highly mutated gene groups and pathways** ( )

When you run the Identify highly mutated gene groups and pathways analysis, you have to specify both the annotation association file, a gene track, and finally which ontology (cellular component, biological process or molecular function) you would like to test for (see figure 28.3).

The analysis starts by associating all of the variants from the input variant file with genes in the

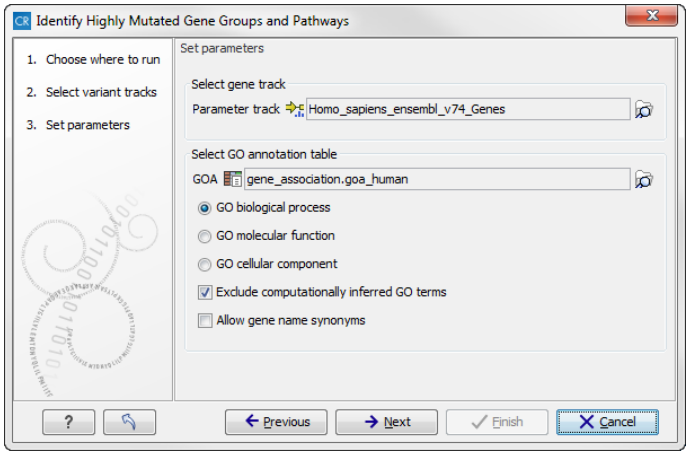


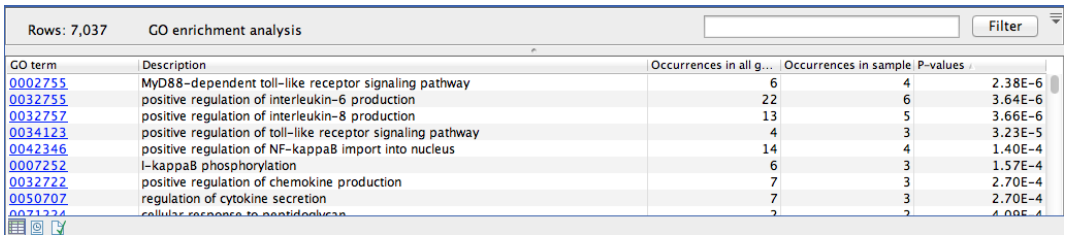
Figure 28.3: *Select gene track, GO annotation table, and ontology.*

gene track, based on overlap with the gene annotations. A variant track can be created with the *Biomedical Genomics Workbench* (variant callers as in section 22.12 or 22.9.

Next, the Workbench tries to match gene names from the gene (annotation) track with the gene names in the GO association file. Please be aware that the same gene name definition should be used in both files.

Based on this, the Workbench finds GO terms that are over-represented in the list. A hypergeometric test is used to identify over-represented GO terms by testing whether some of the GO terms are over-represented in a given gene set, compared to a randomly selected set of genes.

The result is a table with GO terms and the calculated p-value for the candidate variants, and a new variant file with annotated GO terms and the corresponding p-value (see figure 28.4). The p-value is the probability of obtaining a test statistic at least as extreme as the one that was actually observed, or in other words how significant (trustworthy) a result is. In case of a small p-value the probability of achieving the same result by chance with the same test statistic is very small.



GO term	Description	Occurrences in all g...	Occurrences in sample	P-values
0002755	MyD88-dependent toll-like receptor signaling pathway	6	4	2.38E-6
0032755	positive regulation of interleukin-6 production	22	6	3.64E-6
0032757	positive regulation of interleukin-8 production	13	5	3.66E-6
0034123	positive regulation of toll-like receptor signaling pathway	4	3	3.23E-5
0042346	positive regulation of NF-kappaB import into nucleus	14	4	1.40E-4
0007252	I-kappaB phosphorylation	6	3	1.57E-4
0032722	positive regulation of chemokine production	7	3	2.70E-4
0050707	regulation of cytokine secretion	7	3	2.70E-4

Figure 28.4: *The results of the analysis.*

28.3 Identify mutated genes

The Identify Mutated Genes tool can be used if you would like to identify only variants that fall within genes. This is done by comparing a gene track with a variant track holding the variants (mutations) of interest. The variant track is used as an overlap filter for filtering the gene track based on overlap with the variant track.

To run the tool:

Toolbox | **Identify Candidate Genes** ( ) | **Identify Mutated Genes** ( )

Select the gene track you wish to filter and click on the button labeled **Next** (see figure 28.5).

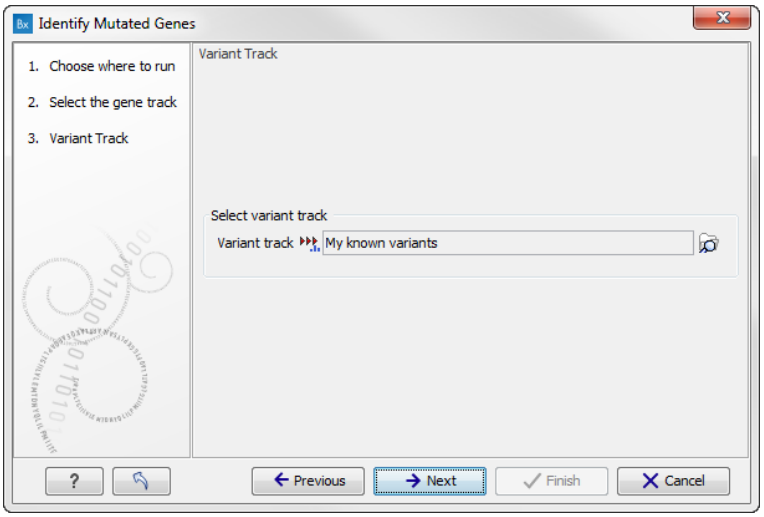


Figure 28.5: *Select the variant track to be used for comparison.*

Next, select the variant track that should be used for comparison. An overlap has a simple definition – if the annotation used as input has at least one shared position with the other track, there is an overlap. The boundaries of the annotations do not need to match.

28.4 Select genes by name

The name filter allows you to use a list of names as input to create a new track only with these names. This is useful if you wish to filter your variants so that only those within certain genes are reported.

Toolbox | **Identify Candidate Genes** ( ) | **Select Genes By Name** ( )

Select the track you wish to filter and click **Next**.

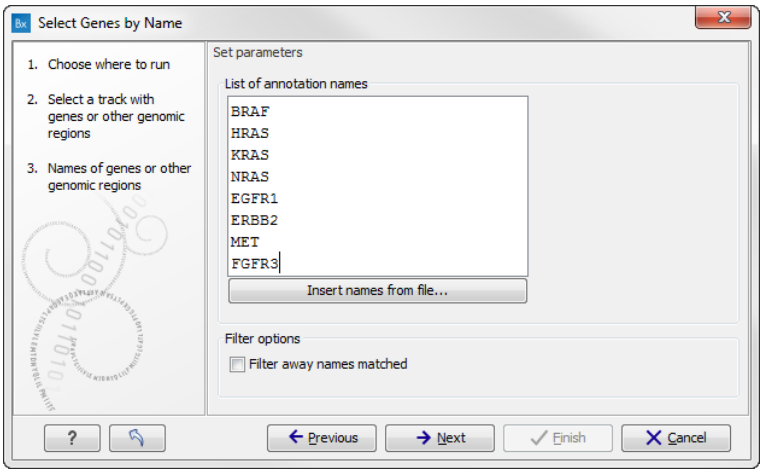


Figure 28.6: *Specify names for filtering.*

As shown in figure 28.6, you can specify a list of annotation names. Each name should be on a separate line.

In the bottom part of the wizard you can choose whether you wish to keep the annotations that

are found, or whether you wish to exclude them. In the use case described above a track was created with only those annotations being kept that matched the specified names. Sometimes the other option may be useful, for example if you wish to screen certain categories of genes from the analysis (for example excluding all cancer genes to reduce the risk of coincidental findings when analyzing patient samples).

Part VIII

Transcriptomic analysis

Chapter 29

RNA-Seq Analysis tools

Contents

Based on an annotated reference genome, the *Biomedical Genomics Workbench* supports **RNA-Seq Analysis** by mapping next-generation sequencing reads and distributing and counting the reads across genes and transcripts. Subsequently, the results can be used for expression analysis. The tools from the RNA-Seq folder automatically account for differences due to sequencing depth, removing the need to normalize input data. The statistical analysis and visualization tools of the RNA-Seq folder make extensive use of the metadata system.

TMM Normalization

Since the sequencing depth might differ between samples, a per-sample library size normalization must be performed before samples can be compared. In the case of the tools included in the RNA-Seq folder, this normalization is automatically applied by the tools.

All of the tools in the RNA-Seq folder use the TMM (trimmed mean of M values) normalization method [Robinson and Oshlack, 2010] to calculate effective libraries sizes, which are then used as part of the per-sample normalization. TMM normalization is the normalization used in EdgeR [Robinson et al., 2010].

TMM normalization adjusts library sizes based on the assumption that most genes are not differentially expressed. Therefore, it is important not to make subsets of the count data before doing statistical analysis or visualization, as this can lead to differences being normalized away.

For the expression visualization tools (Create Heat Map and PCA for RNA-Seq) additional filtering and normalization are performed:

- 'log CPM' (Counts per Million) values are calculated for each gene. The CPM calculation uses the effective library sizes as calculated by the TMM normalization.
- After this first normalization, a second one is performed across samples for each gene: the counts for each gene are mean centered, and scaled to unit variance.
- Genes or transcripts with zero expression across all samples or invalid values (NaN or +/- Infinity) are removed.

Metadata for RNA-Seq

The statistical analysis and visualization tools of the RNA-Seq folder make extensive use of the metadata system. For example, metadata are required when defining the experimental design in the Differential Expression for RNA-Seq tool, and can be used to add extra layers of insight in the Create Heat Map and PCA for RNA-Seq tools.

To get the most out of these tools we recommend that all input expression tracks have associated metadata, as shown in figure 29.1. For information about how to use and setup metadata, please see section 3.2.

29.1 RNA-Seq analysis

The following describes the overall process of the RNA-Seq analysis when using an annotated eukaryote genome. See section 29.1.1 for more information on other types of reference data.

The RNA-Seq analysis is done in several steps: First, all annotated transcripts are extracted (using an *mRNA* track). If there are several annotated splice variants, they are all extracted.

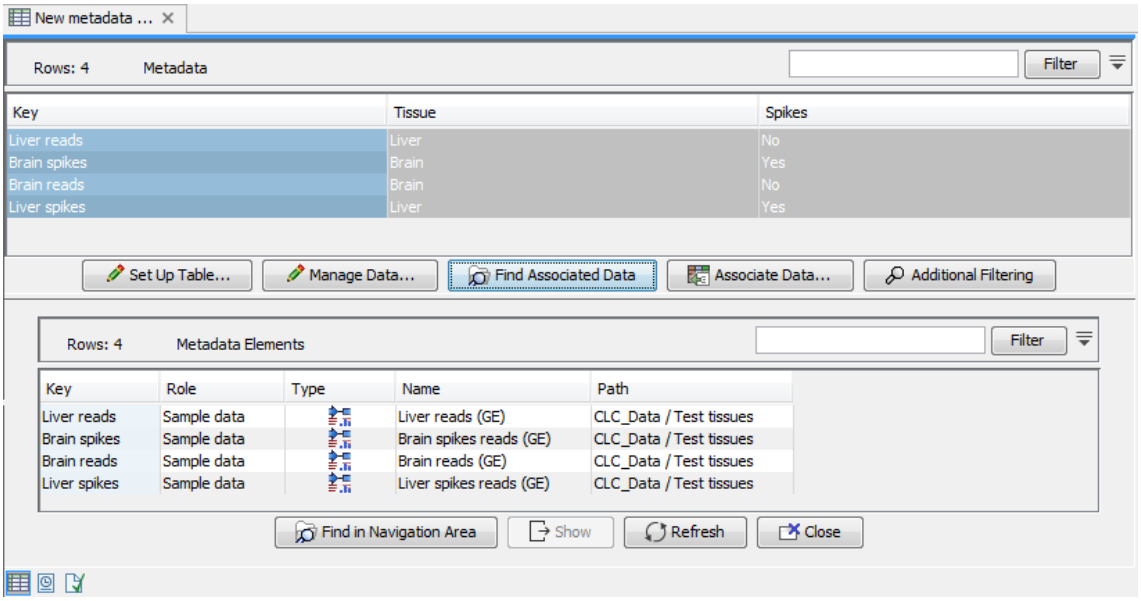


Figure 29.1: *An example of expression tracks with associated metadata.*

An example is shown in figure 29.2.

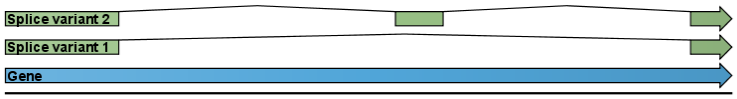


Figure 29.2: *A simple gene with three exons and two splice variants.*

This is a simple gene with three exons and two splice variants. The transcripts are extracted as shown in figure 29.3.

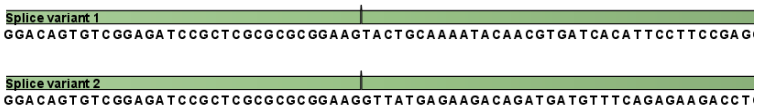


Figure 29.3: *All the exon-exon junctions are joined in the extracted transcript.*

Next, the reads are mapped against all the transcripts plus to the whole genome. For more information about the read mapper, see section 22.1.

From this mapping, the reads are categorized and assigned to the genes using the EM estimation algorithm , and expression values for each gene and each transcript are calculated.

29.1.1 Specifying reads and reference

To start the RNA-Seq analysis, go to:

Toolbox | **RNA-Seq Analysis ()** | **RNA-Seq Analysis ()**

This opens a dialog where you select the **sequencing reads**. Note that you need to import the sequencing data into the Workbench before it can be used for analysis. Importing read data is described in section 6.3.

If you have several samples that you wish to analyze independently and compare afterwards, you can run the analysis in batch mode (see section 8.3).

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

You are now presented with the dialog shown in figure 29.4.

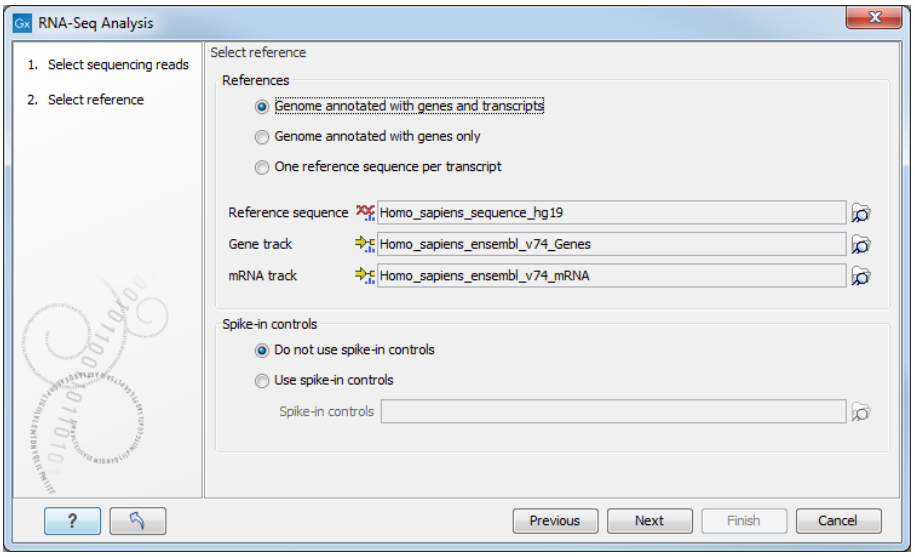


Figure 29.4: *Defining a reference genome for RNA-Seq.*

At the top, there are three options concerning how the reference sequences are annotated.

- **Genome annotated with genes and transcripts**. This option is the only option where splicing is taken into account. When this option is selected, both a *Gene* and an *mRNA* track should be provided in the boxes below. The mRNA annotations are used to define how the transcripts are spliced (as shown in figure 29.2). The reference sequence, gene, and mRNA tracks are provided with the Biomedical Genomics Workbench and can be downloaded using the **Data Management** ( ) function found in the top right corner of the Workbench (see section 13.1).

  When using this option, Expression values, RPKM and TPM are calculated based on the lengths of the transcripts provided by the mRNA track. If a gene's transcript annotation is absent from the mRNA track, all values will be set to 0 unless the option "Calculate expression for genes without transcript" is checked in a later dialog.

- **Genome annotated with genes only**. This option should be used for in situations where you are not interested in transcript level expression. When this option is selected, a *Gene* track should be provided in the box below.

  When using this option, Expression values, RPKM and TPM are calculated based on the lengths of the genes provided by the Genes track.

- **One reference sequence per transcript**. This option is suitable for situations where the reference is a list of sequences. Each sequence in the list will be treated as a "transcript" and expression values are calculated for each sequence. This option is most often used if the reference is a product of a *de novo* assembly of RNA-Seq data. When this option is selected, only the reference sequence should be provided, either as a sequence track or a sequence list. Expression values, RPKM and TPM are calculated based on the lengths of sequences from the sequence track or sequence list.

At the bottom of the dialog you can choose between these two options:

- **Do not use spike-in controls**.
- **Use spike-in controls**. In this case, you can provide a spike-in control file in the field situated at the bottom of the dialog window. Make sure you remember to check the option to output a report in the last wizard step, as the report is the only place where the spike-in controls results will be available. During analysis, the spike-in data is added to the references. However, all traces of having used spike-ins are removed from the output tracks.

If spike-ins have been used, the quality control results are shown in the output report. So when using spike-in, make sure that the option to output a report is checked.

To learn how to import spike-in control files, see section 6.4.

29.1.2 Defining mapping options for RNA-Seq

When the reference has been defined, click **Next** and you are presented with the dialog shown in figure 29.5.

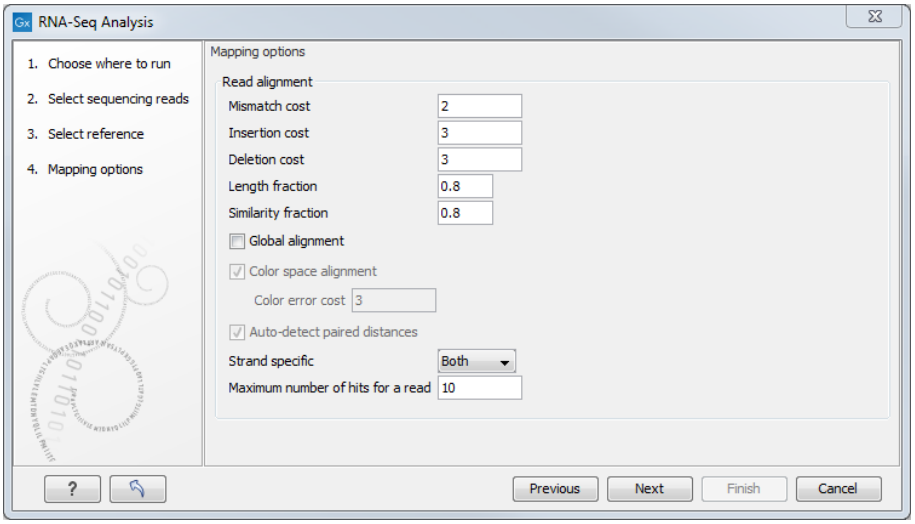


Figure 29.5: *Defining mapping parameters for RNA-Seq.*

The mapping parameters are identical to those applying to **Map Reads to Reference**, as the underlying mapping is performed in the same way. For a description of the parameters, please see section 22.1.3.

For the estimation of paired reads distances, RNA-Seq uses the transcript level reference sequence information. This means that introns are not included in the distance measurement. The paired distance measurement will only include transcript sequence, reflecting the true nature of the sequence on which the paired reads were produced.

In addition to the generic mapping parameters, two RNA-Seq specific parameters can be set:

- **Maximum number of hits for a read**. A read that matches equally well to more *distinct* places in the references than the 'Maximum number of hits for a read' specified will not be

mapped (the notion of *distinct* places is elaborated below). If a read matches to multiple distinct places, but less than the specified maximum number, it will be randomly assigned to one of these places. The random distribution is done by the EM algorithm (see section 29.1.3)

The definition of a *distinct* place in the references is complicated because each annotated transcript is extracted and used as reference for the read mapping (if the "Genome annotated with genes and transcripts" is selected in figure 29.4). To exemplify, consider a gene with 10 transcripts and 11 exons, where all transcripts have exon 1, and each of the 10 transcripts have only one of the exons 2 to 11. Exon 1 will be represented 11 times in the references (once for the gene region and once for each of the 10 transcripts). Reads that match to exon 1 will thus match to 11 of the extracted references. However, when the mappings are considered in the coordinates of the main reference genome, it becomes evident that the 11 match places are not distinct but in fact identical. In this case this will just count as one distinct placement of the read, and it will *not* be discarded for exceeding the maximum number of hits limit. Similarly, when a multi-match read is randomly assigned to one of its match places, each distinct place is considered only once.

The limit for how many non-specific matches a read is allowed to have is applied first to the set of gene matches (if any), and then to intergenic matches. As an example using the default value of 10, if a read matches equally well 8 places within genes and 50 places in intergenic regions, it is still considered a valid match. It will only be discarded if the number of matches within genes is above the limit, or if there are no gene matches at all and the number of intergenic matches exceeds the limit.

Note that, although a read is mapped *distinctly* at the gene level, it does not necessarily map *uniquely* to a particular transcript of the gene. The above example with a gene with 10 transcripts and 11 exons, where all transcripts have exon 1, and each of the 10 transcripts have only one of the exons 2 to 11, is a good and easy to understand example of this: all reads that are mapped to exon 1 are *uniquely* mapped at the gene level but are *non-specific matches* at the transcript level. A more complicated example is that you may have a gene with transcript annotations where one transcript has a longer version of an exon than the other. In this case you may have reads that may either be mapped entirely within the long version of the exon, or across the exon-exon boundary of one of the transcripts with the short version of the exon. Such an example is provided by the gene 'Ftl1' in the example below (gene and mRNA annotations for that gene are shown in figure 29.6, along with the reads mapping to the gene).

When you zoom in on the regions at the end of the second exons and the beginning of the third exons (Figure 29.7) you see that the reference sequence is identical in the start of the part of the second exons that is only present in the long version, and in the start of the third exons (they share the sequence 'CTGCACA'). So a read that is '...TCATCTTGAGATGGCTTCTGCACA' may be either mapped entirely within the long version of the second exons, or across the exon-exon boundary of the short version of the second exon and the third exon. For reporting expression levels at the transcript level, reads are assigned among the transcripts to which they map by the Expectation Maximization algorithm.

- **Strand-specific alignment**. When this option is checked, the user can specify whether the reads should be mapped in the same orientation as the transcript from which they originate (forward) or in the reverse direction (reverse). This will typically be appropriate when a strand specific protocol for read generation has been used. It allows assignment of the

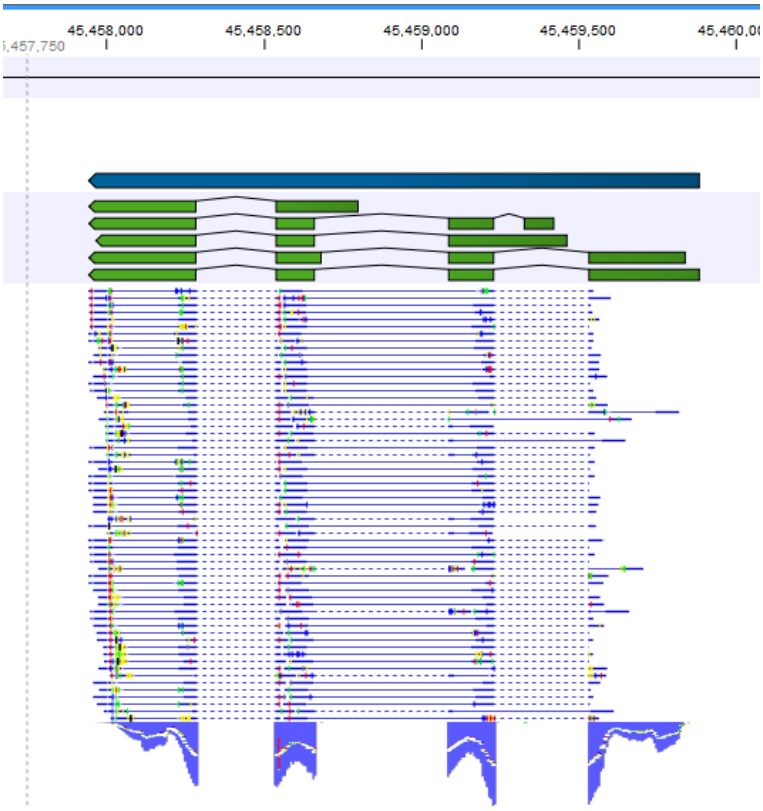


Figure 29.6: *The gene 'Ftl1' from the mouse chromosome 7.*

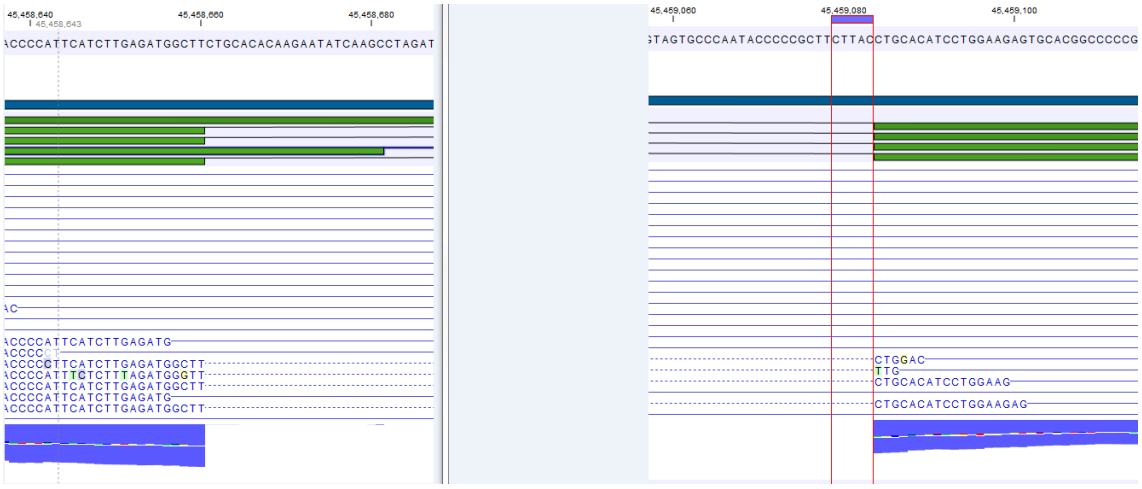


Figure 29.7: *The regions at the end of the second exons and the beginning of the third exons of the mRNA transcripts for the gene 'Ftl1'.*

reads to the right gene in cases where overlapping genes are located on different strands. Without the strand-specific protocol, this would not be possible (see [Parkhomchuk et al., 2009]). Note that when running RNA seq with the strand specific option turned on you can only make use of pairs in forward-reverse orientation, meaning that mate pairs are not supported.

29.1.3 The EM estimation algorithm

The EM estimation algorithm is inspired by the RSEM and eXpress methods. It iteratively estimates the abundance of transcripts, and assigns reads to transcripts according to these abundances. To illustrate the interpretation of 'transcript abundance', consider the following two examples.

In the first example, we have a gene with two transcripts, where one transcript is twice as long as the other (figure 29.8). This longer transcript is also twice as abundant, meaning that in the exon common to both transcripts two of the three reads come from the longer transcript, and the final read comes from the shorter transcript. The longer transcript has a second exon which also generates two reads. We see then that the longer transcript is twice as abundant, but because it is twice as long it generates four times as many reads.

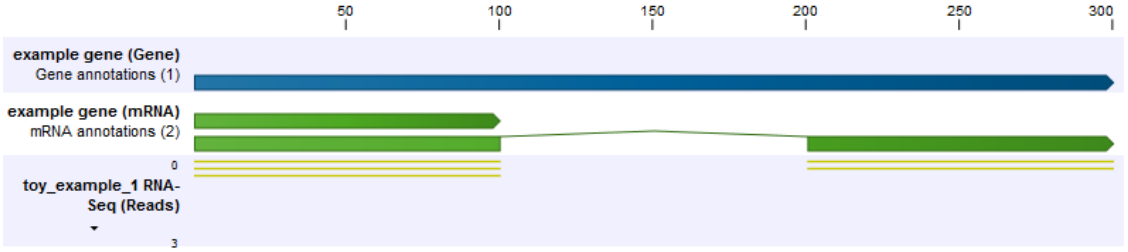


Figure 29.8: *The longer transcript has twice the abundance, but four times the number of reads as the shorter transcript.*

In the second example, the set up is the same, but now the shorter transcript is twice as abundant as the longer transcript (figure 29.9). Because the longer transcript is twice as long, there are equal numbers of reads from each transcript.

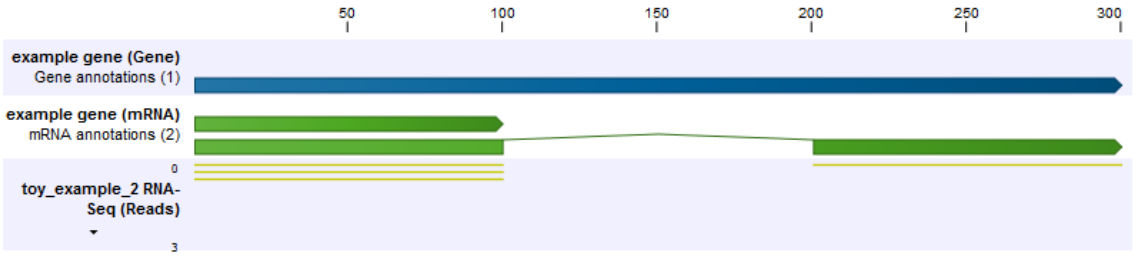


Figure 29.9: *The longer transcript has half the abundance, but the same number of reads as the shorter transcript.*

To estimate the transcript abundances, we carry out an expectation-maximization procedure. Before explaining the procedure, first we define the concept of a mapping. A mapping is a set of transcripts, to which a read may map. In the above examples, some reads have the mapping $a_1 = \{t_1, t_2\}$ (these are non-uniquely mapping reads), and some reads have the mapping $a_2 = \{t_2\}$ (these are 'uniquely' mapping reads). In both examples, the count of mapping a_1 is 3, because there are 3 shared reads between the transcripts. The count of mapping a_2 is 2 in the first example, and 1 in the second example.

The expectation-maximization algorithm proceeds as follows:

1. The transcript abundances are initialized to the uniform distribution, i.e. at the start all transcripts are assumed to be equally expressed.

2. Expectation step: the current (assumed) transcript abundances are used to calculate the expected count of each transcript, i.e. the number of reads we expect should be assigned to the given transcript. This is done by looping over all mappings that include the given transcript, and assigning a proportion of the total count of that mapping to the transcript. The proportion corresponds to the proportion of the total transcript abundance in the mapping that is due to the target.

3. Maximization step: the currently assigned counts of each transcript are used to re-compute the transcript abundances. This is done by looping over all targets, and for each target, dividing the proportion of currently assigned counts for the transcript (=total counts for transcript/total number of reads) by the target length. This is necessary because longer transcripts are expected to generate proportionally more reads.

4. Repeat from step 2 until convergence.

Below, we illustrate how the expectation-maximization algorithm converges to the expected abundances for the above two examples.

```
Example 1:
Initially: transcript 2 abundance = 0.50, count: 0.00, transcript 1 abundance = 0.50, count: 0.00
After 1 round: transcript 2 abundance = 0.54, count: 3.50, transcript 1 abundance = 0.46, count: 1.50
After 2 rounds: transcript 2 abundance = 0.57, count: 3.62, transcript 1 abundance = 0.43, count: 1.38
After 3 rounds: transcript 2 abundance = 0.59, count: 3.70, transcript 1 abundance = 0.41, count: 1.30
After 4 rounds: transcript 2 abundance = 0.60, count: 3.76, transcript 1 abundance = 0.40, count: 1.24
After 5 rounds: transcript 2 abundance = 0.62, count: 3.81, transcript 1 abundance = 0.38, count: 1.19
After 6 rounds: transcript 2 abundance = 0.62, count: 3.85, transcript 1 abundance = 0.38, count: 1.15
After 7 rounds: transcript 2 abundance = 0.63, count: 3.87, transcript 1 abundance = 0.37, count: 1.13
After 8 rounds: transcript 2 abundance = 0.64, count: 3.90, transcript 1 abundance = 0.36, count: 1.10
After 9 rounds: transcript 2 abundance = 0.64, count: 3.92, transcript 1 abundance = 0.36, count: 1.08
After 10 rounds: transcript 2 abundance = 0.65, count: 3.93, transcript 1 abundance = 0.35, count: 1.07
After 11 rounds: transcript 2 abundance = 0.65, count: 3.94, transcript 1 abundance = 0.35, count: 1.06
After 12 rounds: transcript 2 abundance = 0.65, count: 3.95, transcript 1 abundance = 0.35, count: 1.05
After 13 rounds: transcript 2 abundance = 0.66, count: 3.96, transcript 1 abundance = 0.34, count: 1.04
After 14 rounds: transcript 2 abundance = 0.66, count: 3.97, transcript 1 abundance = 0.34, count: 1.03
After 15 rounds: transcript 2 abundance = 0.66, count: 3.97, transcript 1 abundance = 0.34, count: 1.03


Example 2:
Initially: transcript 2 abundance = 0.50, count: 0.00, transcript 1 abundance = 0.50, count: 0.00
After 1 round: transcript 2 abundance = 0.45, count: 2.50, transcript 1 abundance = 0.55, count: 1.50
After 2 rounds: transcript 2 abundance = 0.42, count: 2.36, transcript 1 abundance = 0.58, count: 1.64
After 3 rounds: transcript 2 abundance = 0.39, count: 2.26, transcript 1 abundance = 0.61, count: 1.74
After 4 rounds: transcript 2 abundance = 0.37, count: 2.18, transcript 1 abundance = 0.63, count: 1.82
After 5 rounds: transcript 2 abundance = 0.36, count: 2.12, transcript 1 abundance = 0.64, count: 1.88
After 6 rounds: transcript 2 abundance = 0.35, count: 2.08, transcript 1 abundance = 0.65, count: 1.92
After 7 rounds: transcript 2 abundance = 0.35, count: 2.06, transcript 1 abundance = 0.65, count: 1.94
After 8 rounds: transcript 2 abundance = 0.34, count: 2.04, transcript 1 abundance = 0.66, count: 1.96
After 9 rounds: transcript 2 abundance = 0.34, count: 2.03, transcript 1 abundance = 0.66, count: 1.97
After 10 rounds: transcript 2 abundance = 0.34, count: 2.02, transcript 1 abundance = 0.66, count: 1.98
After 11 rounds: transcript 2 abundance = 0.34, count: 2.01, transcript 1 abundance = 0.66, count: 1.99
After 12 rounds: transcript 2 abundance = 0.34, count: 2.01, transcript 1 abundance = 0.66, count: 1.99
After 13 rounds: transcript 2 abundance = 0.33, count: 2.01, transcript 1 abundance = 0.67, count: 1.99
After 14 rounds: transcript 2 abundance = 0.33, count: 2.00, transcript 1 abundance = 0.67, count: 2.00
After 15 rounds: transcript 2 abundance = 0.33, count: 2.00, transcript 1 abundance = 0.67, count: 2.00
```

Once the algorithm has converged, every non-uniquely mapping read is assigned randomly to a particular transcript according to the abundances of transcripts within the same mapping. The total transcript reads column reflects these assignments. The RPKM and TPM values are then computed from the counts assigned to each transcript.

29.1.4 Calculating expression values from RNA-Seq

When the reference has been defined, click **Next** and you are presented with the dialog shown in figure 29.10.

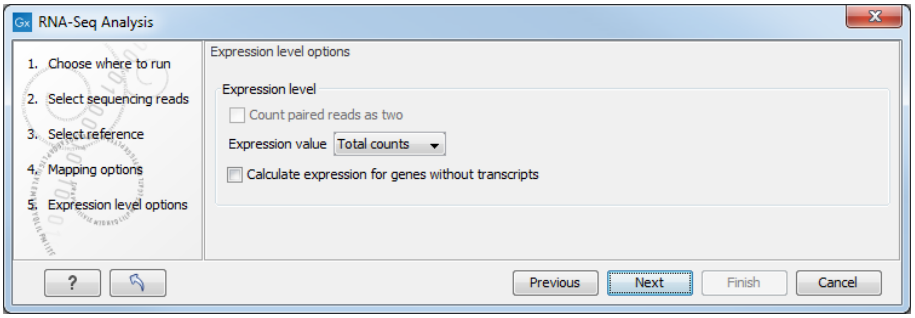


Figure 29.10: *Defining how expression values should be calculated.*

These parameters determine the way expression values are counted.

Count paired reads as two

The *Biomedical Genomics Workbench* supports the direct use of paired data for RNA-Seq. A combination of single reads and paired reads can also be used. There are three major advantages of using paired data:

- Since the mapped reads span a larger portion of the reference, there will be fewer non-specifically mapped reads. This means that generally there is a greater accuracy in the expression values.
- This in turn means that there is a greater chance of accurately measuring the expression of transcript splice variants. As single reads (especially from the short reads platforms) typically only span one or two exons, many cases will occur where expression of splice variants sharing the same exons cannot be determined accurately. With paired reads, more combinations of exons will be identified as being unique for a particular splice variant.[1]
- It is possible to detect **Gene fusions** when one read in a pair maps in one gene and the other part maps in another gene. Several reads exhibiting the same pattern supports the presence of a fusion gene.

You can read more about how paired data are imported and handled in section 6.3.7.

When counting the mapped reads to generate expression values, the *Biomedical Genomics Workbench* needs to be told how to handle the counting of paired reads. The default behavior of the *Biomedical Genomics Workbench* is to count fragments (FPKM) rather than individual reads when two reads map as an intact pair. That is, an intact pair is given a count of one. Reads from a pair are considered part of a broken pair when the reads map outside the estimated pair distance, map in the wrong orientation, or only one of the reads of the pair maps. Neither member of a broken pair is counted when the default counting scheme is used. The reasoning is that when reads map as a broken pair, it is an indication that something is not right. For

[1] Note that the *Biomedical Genomics Workbench* only calculates the expression of the transcripts already annotated on the reference.

example, perhaps the transcripts are not represented correctly on the reference or there are errors in the data. In general, more confidence can be placed on an intact pair representing transcription within the sample. If a combination of paired and single reads are input into the analysis, then single reads that map are given a count of one. This is different from reads input into the analysis as part of a pair, but where their partner did not map.

In some situations it may be too strict to disregard broken pairs as is done using the default counting scheme. This could be the case where there is a high degree of variation in the sample compared to the reference or where the reference lacks comprehensive transcript annotations. By checking the **Count paired reads as two** option, you choose to count mapped 'reads' (RPKM) rather than mapped 'fragments' (FPKM). That means that, the two reads in an intact pair are each counted as one mapped read (so an intact pair contributes with a total count of two), and mapped members of broken pairs will each get given a count of one. Single mapped reads are also given a count of one. Note that this approach does not represent the abundance of fragments being sequenced correctly, since the two reads of a pair derive from the same fragment, whereas a fragment sequenced with single reads only give rise to one read.

Note that whether you choose to calculate RPKM or FPKM, the value will be given in a column called "RPKM" for all subsequent analysis.

Expression value

Please note that reads that map outside genes are counted as intergenic hits only and thus do not contribute to the expression values. If a read maps equally well to a gene and to an inter-genic region, the read will be placed in the gene.

The expression values are created on two levels as two separate result files: one for genes and one for transcripts (if the "Genome annotated with genes and transcripts" is selected in figure 29.4). The content of the result files is described in section 29.1.5.

The **Expression value** parameter describes how expression per gene or transcript can be defined in different ways on both levels:

- **Total counts**. When the reference is annotated with genes only, this value is the total number of reads mapped to the gene. For un-annotated references, this value is the total number of reads mapped to the reference sequence. For references annotated with transcripts and genes, the value reported for each gene is the number of reads that map to the exons of that gene. The value reported per transcript is the total number of reads mapped to the transcript.
- **Unique counts**. This is similar to the above, except only reads that are uniquely mapped are counted (read more about the distribution of non-specific matches in section 29.1.2).
- **TPM**. (Transcripts per million). This is computed as $\frac{\text{RPKM} \cdot 10^6}{\sum \text{RPKM}}$, where the sum is over the RPKM values of all genes/transcripts.
- **RPKM**. This is a normalized form of the "Total counts" option (see more in section 29.1.4).

Please note that all values are present in the output. The choice of expression value only affects how Expression Tracks are visualized in the track view but the results will not be affected by this choice as the most appropriate expression value is automatically selected for the analysis

being performed: for detection of differential expression this is the "Total counts" value, and for the other tools this is a normalized and transformed version of the "Total counts" as described below.

Calculate expression for genes without transcripts

For genes without annotated transcripts, the RPKM cannot be calculated since the total length of all exons is needed. By checking the **Calculate expression for genes without transcripts**, the length of the gene will be used in place of an "exon length". If the option is not checked, there will be no RPKM value reported for those genes.

Definition of RPKM RPKM, Reads Per Kilobase of exon model per Million mapped reads, is defined in this way [Mortazavi et al., 2008]:

$$RPKM = \frac{total\ exon\ reads}{mapped\ reads(millions) \times exon\ length\ (KB)}.$$

For prokaryotic genes and other non-exon based regions, the calculation is performed in this way:

$$RPKM = \frac{total\ gene\ reads}{mapped\ reads(millions) \times gene\ length\ (KB)}.$$

Total exon reads This value can be found in the column with header **Total exon reads** in the expression track. This is the number of reads that have been mapped to exons (either within an exon or at the exon junction). When the reference genome is annotated with gene and transcript annotations, the mRNA track defines the exons, and the total exon reads are the reads mapped to all transcripts for that gene. When only genes are used, each gene in the gene track is considered an exon. When an un-annotated sequence list is used, each sequence is considered an exon.

Exon length This is the number in the column with the header **Exon length** in the expression track, divided by 1000. This is calculated as the sum of the lengths of all exons (see definition of exon above). Each exon is included only once in this sum, even if it is present in more annotated transcripts for the gene. Partly overlapping exons will count with their full length, even though they share the same region.

Mapped reads The sum of all mapped reads as listed in the RNA-Seq analysis report. If paired reads were used in the mapping, mapped fragments are counted here instead of reads, unless the **Count paired reads as two** option was selected. For more information on how expression is calculated in this case, see section 29.1.4.

29.1.5 Specifying RNA-Seq outputs

Clicking **Next** will allow you to specify the output options as shown in figure 29.11.

The main results of the RNA-Seq analysis are:

- **Expression Tracks** One track summarizing expression at the gene level is produced. The track name ends in **(GE)**. If the "Genome annotated with genes and transcripts" option

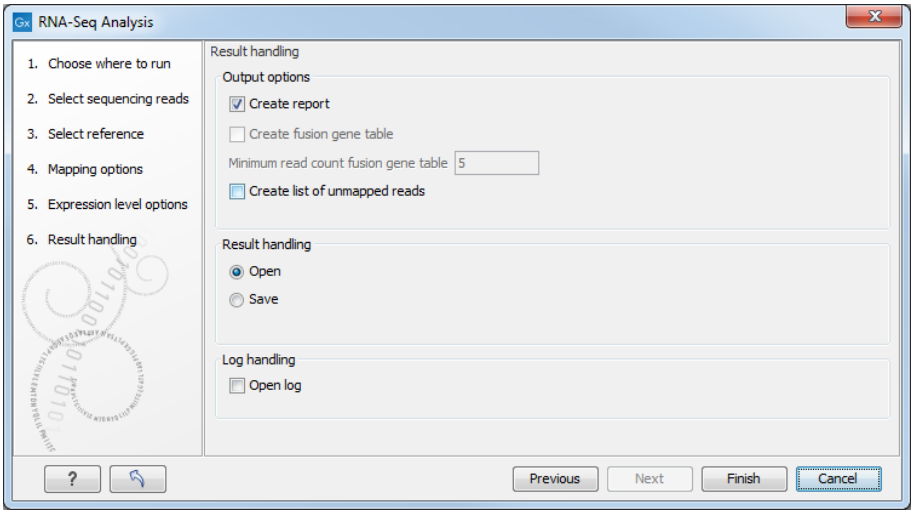


Figure 29.11: *Selecting the output of the RNA-Seq analysis.*

was selected, as shown in figure 29.4, then a second track summarizing expression at the transcript level is also produced. This track has a name ending with **(TE)**.

- **Reads track** This track contains the mapping of the reads to the references. This track has a name ending with **(Reads)**.

In addition, the following optional results can be selected:

- **Create list of unmapped reads**. Creates a list of the reads that either did not map to the reference at all or that were non-specific matches with more placements than specified (see section 29.1.2). If you started with paired reads then more than one list of unmapped reads may be produced: paired reads are put in one list, with a name that ends in (paired), singe reads, including members of broken pairs, are put in a read list with a name than ends in (single).
- **Create report**. Creates a report of the results. See **RNA-Seq report** below for a description of the information contained in the report. This report is also the only place results of the spike-in controls will be available.
- **Create fusion gene table**. An option that is enabled when using paired data. Creates a table that lists potential fusion genes. This, along with the **Minimum read count**, is described further below in section **Gene fusion reporting**.

29.1.6 Expression tracks

Both tracks can be shown in a **Table** ( ) and a **Graphical** ( ) view.

The expression track table view has three button (figure 29.12).

- The "Create track from Selection" will create a Track using selected rows.
- The "Select Genes in Other Views" button finds and selects the currently selected genes and transcripts in all other open expression track table views.

- The "Copy Gene Names to Clipboard" button copies the currently selected gene names to the clipboard.

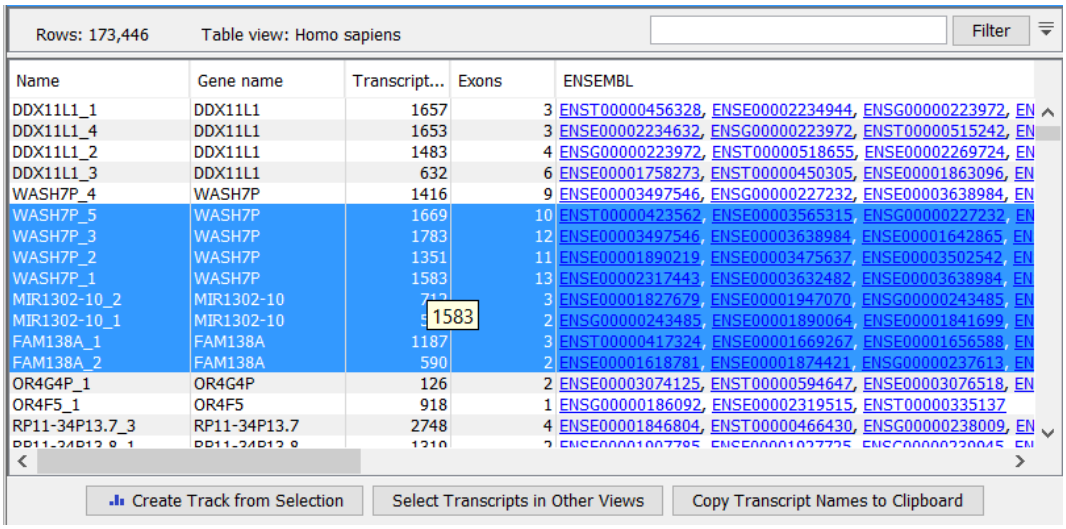


Figure 29.12: *RNA-Seq results shown in a table view.*

By creating a **Track list**, the graphical view can be shown together with the read mapping track and tracks from other samples:

File | **New** | **Genome Browser View (** **)**

Select the mapping and expression tracks of the samples you wish to visualize together and select the annotation tracks used as reference for the RNA-Seq and click **Finish**.

Once the track list is shown, double-click the label of the expression track to show it in a table view.

Clicking a row in the table makes the track list view jump to that location, allowing for quick inspection of interesting parts of the RNA-Seq read mapping (see an example in figure 29.13).

Reads spanning two exons are shown with a dashed line between each end as shown in figure 29.13, and the thin solid line represents the connection between two reads in a pair.

When doing comparative analysis, double click on one of the Expression or Statistical Comparison tracks in a track list to get its table view. Then click on the "Select genes in other views" button in any other table or expression browser will cause the track list to scroll to the selected gene.

Expression tracks can also be used to annotate variants using the **Annotate with Overlap Information** tool. Select the variant track as input and annotate with the expression track. For variants inside genes or transcripts, information will be added about expression (counts, expression value etc) from the gene or transcript in the expression track. Read more about the annotation tool in section 25.1.

Gene-level expression The gene-level expression track (GE) holds information about counts and expression values for each gene. It can be opened in a **Table view** ( ) allowing sorting and filtering on all the information in the track (see figure 29.14 for an example subset of an expression track).

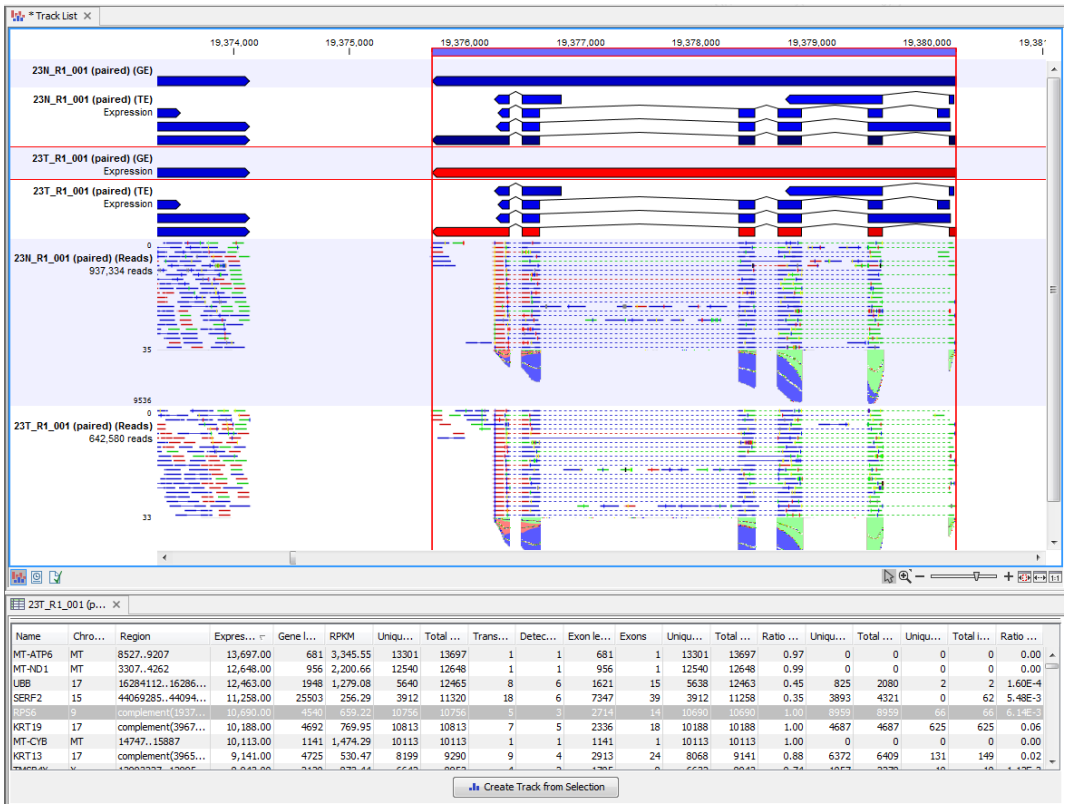
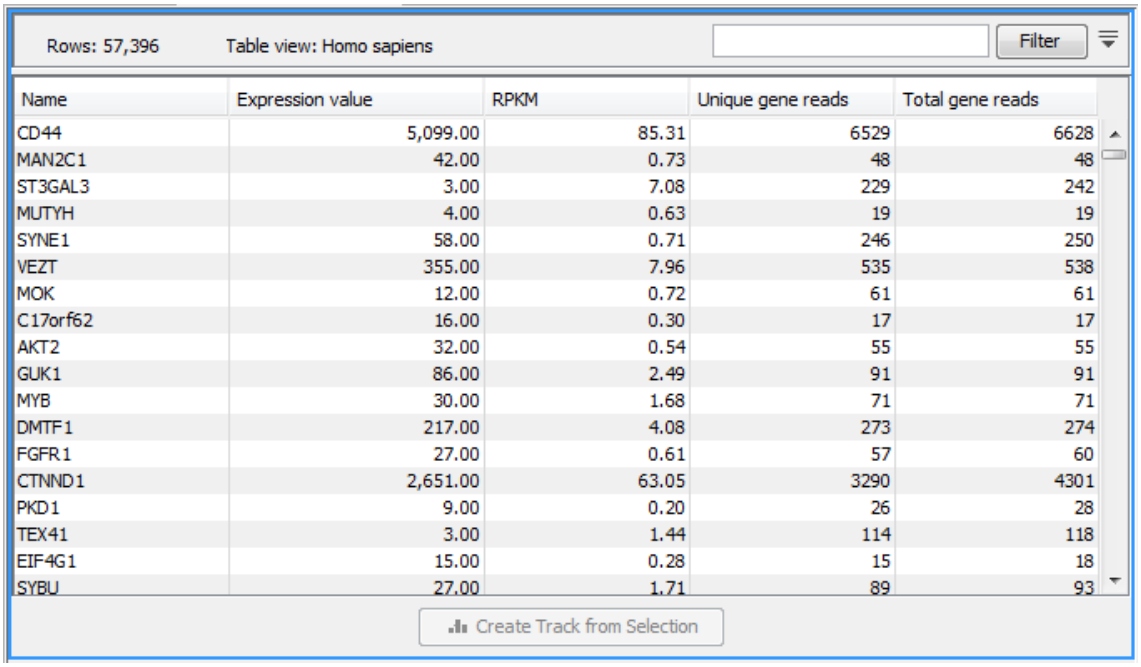


Figure 29.13: *RNA-Seq results shown in a split view with an expression track at the bottom and a track list with read mappings of two samples at the top.*

Each row in the table corresponds to a gene (or reference sequence, if the **One reference sequence per transcript** option was used). The corresponding counts and other information is shown for each gene:

- **Name**. This is the name of the gene or the reference sequence, if the **One reference sequence per transcript** is used.
- **Chromosome and region**. The position of the gene on the genome.
- **Expression value**. This is based on the expression measure chosen as described in section 29.1.4.
- **Gene length** The length of the gene as annotated.
- **TPM (Transcripts per million)**. This is computed as $\frac{\text{RPKM} \cdot 10^6}{\sum \text{RPKM}}$, where the sum is over the RPKM values of all genes/transcripts (see http://bioinformatics.oxfordjournals.org/content/26/4/493.long).
- **RPKM**. This is the expression value measured in RPKM [Mortazavi et al., 2008]: $RPKM = \frac{\textit{total exon reads}}{\textit{mapped reads(millions)} \times \textit{exon length (KB)}}$. See section 29.1.4 for a detailed definition.
- **Unique gene reads**. This is the number of reads that match uniquely to the gene or its transcripts.
- **Total gene reads**. This is all the reads that are mapped to this gene - both reads that map uniquely to the gene or its transcripts and reads that matched to more positions in the



Name	Expression value	RPKM	Unique gene reads	Total gene reads
CD44	5,099.00	85.31	6529	6628
MAN2C1	42.00	0.73	48	48
ST3GAL3	3.00	7.08	229	242
MUTYH	4.00	0.63	19	19
SYNE1	58.00	0.71	246	250
VEZT	355.00	7.96	535	538
MOK	12.00	0.72	61	61
C17orf62	16.00	0.30	17	17
AKT2	32.00	0.54	55	55
GUK1	86.00	2.49	91	91
MYB	30.00	1.68	71	71
DMTF1	217.00	4.08	273	274
FGFR1	27.00	0.61	57	60
CTNND1	2,651.00	63.05	3290	4301
PKD1	9.00	0.20	26	28
TEX41	3.00	1.44	114	118
EIF4G1	15.00	0.28	15	18
SYBU	27.00	1.71	89	93

Figure 29.14: *A subset of a result of an RNA-Seq analysis on the gene level. Not all columns are shown in this figure*

reference (but fewer than the 'Maximum number of hits for a read' parameter) which were assigned to this gene.

- **Transcripts annotated**. The number of transcripts based on the mRNA annotations on the reference. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).
- **Detected transcripts**. The number of annotated transcripts for which there is unambiguous evidence: at least one read mapped to only this transcript. Note that if a gene has 4 detected transcripts, and 8 undetected transcripts, all 4+8=12 transcripts will have the value "detected transcripts = 4".
- **Exon length**. The total length of all exons (not all transcripts).
- **Exons**. The total number of exons across all transcripts.
- **Unique exon reads**. The number of reads that match uniquely to the exons (including across exon-exon junctions).
- **Total exon reads**. Number of reads mapped to this gene that fall entirely within an exon or in exon-exon or exon-intron junctions. As for the 'Total gene reads' this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon of this gene.
- **Ratio of unique to total (exon reads)**. The ratio of the unique reads to the total number of reads in the exons. This can be convenient for filtering the results to exclude the ones where you have low confidence because of a relatively high number of non-unique exon reads.

- **Unique exon-exon reads**. Reads that uniquely match across an exon-exon junction of the gene (as specified in figure 29.13). The read is only counted once even though it covers several exons.
- **Total exon-exon reads**. Reads that match across an exon-exon junction of the gene (as specified in figure 29.13). As for the 'Total gene reads' this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon-exon junction of this gene.
- **Unique intron-exon reads**. Reads that uniquely map across an intron-exon boundary of the gene.
- **Total intron reads**. Reads that maps across an intron-exon boundary of the gene.
- **Ratio of intron to total gene reads**. This can be convenient to identify genes with poor or lacking transcript annotations. If one or more exons are missing from the annotations, there will be a relatively high number of reads mapping in the intron.

Transcript-level expression If the "Genome annotated with genes and transcripts" option is selected in figure 29.4, a transcript-level expression track (TE) is also generated.

The track can be opened in a **Table view** () allowing sorting and filtering on all the information in the track. Each row in the table corresponds to an mRNA annotation in the mRNA track used as reference.

- **Name**. This is the name of the transcript, if the **One reference sequence per transcript** is used.
- **Chromosome and region**. The position of the gene on the genome.
- **Expression value**. This is based on the expression measure chosen as described in section 29.1.4.
- **TPM (Transcripts per million)**. This is computed as $\frac{\text{RPKM} \cdot 10^6}{\sum \text{RPKM}}$, where the sum is over the RPKM values of all genes/transcripts (see http://bioinformatics.oxfordjournals.org/content/26/4/493.long).
- **RPKM**. This is the expression value measured in RPKM [Mortazavi et al., 2008]: $RPKM = \frac{\text{total exon reads}}{\text{mapped reads(millions)} \times \text{exon length (KB)}}$. See section 29.1.4 for a detailed definition.
- **Relative RPKM**. The RPKM for the transcript divided by the maximum of the RPKM values among all transcripts of the same gene. This value describes the relative expression of alternative transcripts for the gene.
- **Gene name**. The name of the corresponding gene.
- **Transcript length**. This is the length of the transcript.
- **Exons**. The total number of exons in the transcript.
- **Transcript ID**. The transcript ID is taken from the transcript_id note in the mRNA track annotations and can be used to differentiate between different transcripts of the same gene.

- **Transcripts annotated**. The number of transcripts based on the mRNA annotations on the reference. Note that this is not based on the sequencing data - only on the annotations already on the reference sequence(s).
- **Detected transcripts**. The number of annotated transcripts for which there is unambiguous evidence: at least one read mapped to only this transcript. Note that if a gene has 4 detected transcripts, and 8 undetected transcripts, all 4+8=12 transcripts will have the value "detected transcripts = 4".
- **Unique transcript reads**. This is the number of reads in the mapping for the gene that are uniquely assignable to the transcript.
- **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 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 assignment has been done. As for the assignment of reads among genes, the assignment of reads within a gene but among transcripts, is done by the EM estimation algorithm (section 29.1.3).
- **Ratio of unique to total (transcript reads)**. The ratio of the unique reads to the total number of reads in the transcripts. This can be convenient for filtering the results to exclude the ones where you have low confidence because of a relatively high number of non-unique transcript reads.

Additional information to GE or TE tracks Both GE and TE tables can offer additional information such as hyperlinks to various databases (e.g., ENSEMBL, HGNC (HUGO Gene Nomenclature Committee), RefSeq, GeneID, etc.) In cases where the mRNA track or the gene track provided have biotype information, a biotype column will be added to the table.

29.1.7 Reads track

A track containing the mapped reads is generated. If you have chosen the strand specific option when setting up your analysis, it may be helpful to note that the colors of mapped single reads represent the orientation of the read relative to the reference provided. When a gene track is provided along with the reference genome, the reads will be mapped using the strand you specified, but the coloring of the read will be relative to the reference gene. If the reads matches the orientation of the gene it is colored green, and if it is opposite to the orientation of the gene it is colored red. A summary list of the colors to expect with different combinations of gene orientation and strand specific mapping options is:

- Strand specific, forward orientation chosen + gene on plus strand of reference = single reads colored green.
- Strand specific, forward orientation chosen + gene on minus strand of reference = single reads colored red.
- Strand specific, reverse orientation chosen + gene on plus strand of reference = single reads colored red.
- Strand specific, reverse orientation chosen + gene on minus strand of reference = single reads colored green.

See Figure 29.15 for an example of forward and reverse reads mapped to a gene on the plus strand.

Note: Reads mapping to intergenic regions will not be mapped in a strand specific way.

Although paired reads are coloured blue, they can be viewed as red and green 'single' reads by selecting the **Disconnect paired reads** box, within the Read Mapping Settings bar on the right-hand side of the track.

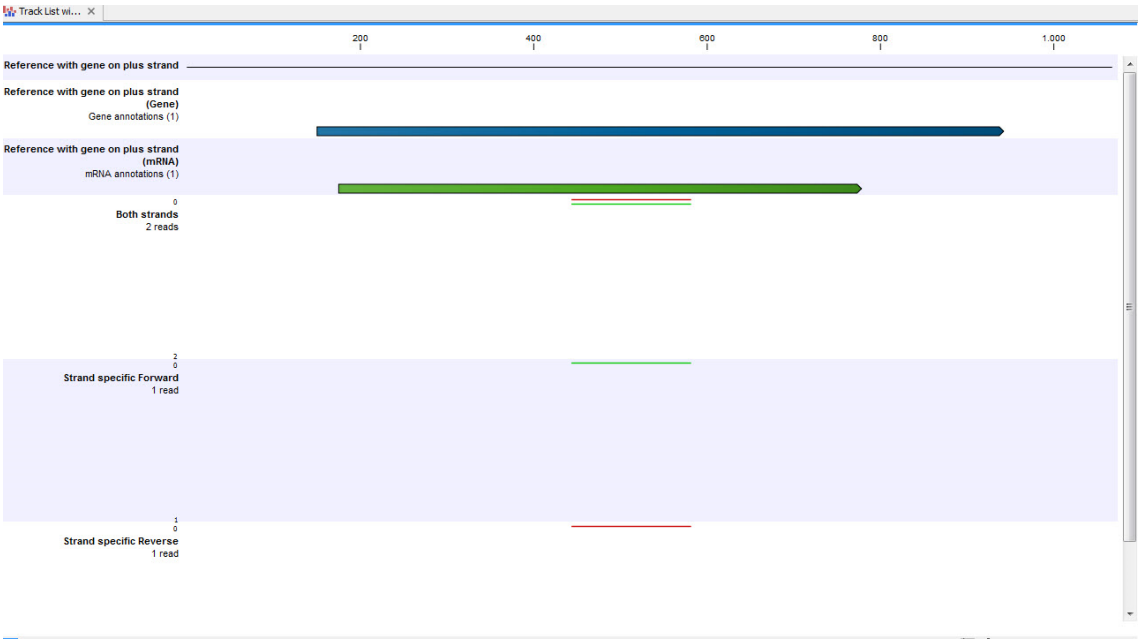


Figure 29.15: *A track list showing a gene and transcript on the plus strand, and various mapping results. The first reads track shows a mapping of two reads (one 'forward' and one 'reverse') using strand specific 'both' option. Both reads map successfully; the forward read coloured green (because it matches the direction of the gene), and the reverse read coloured red. The second reads track shows a mapping of the same reads using strand specific ('forward') option. The reverse read does not map because it is not in the correct direction, therefore only the green forward read is shown. The final reads track shows a mapping of the same reads again but using strand specific 'reverse' option. This time, the green forward read does not map because it is in the wrong direction, and only the red reverse read is shown.*

29.1.8 RNA-Seq report

An example of an RNA-seq report generated if you choose the **Create report** option is shown in figure 29.16.

The report is a collection of the sections described below, some sections included only based on the input provided when starting the tool. If a section is flagged with a pink highlight, it means that something has almost certainly gone wrong in the sample preparation or analysis. A warning message tailored to the highlighted section is added to the report to help troubleshoot the issue. The report can be exported in PDF or Excel format.

Selected input sequences Information about the sequence reads provided as input, including the number of reads in each sample, as well as information about the reference sequences used

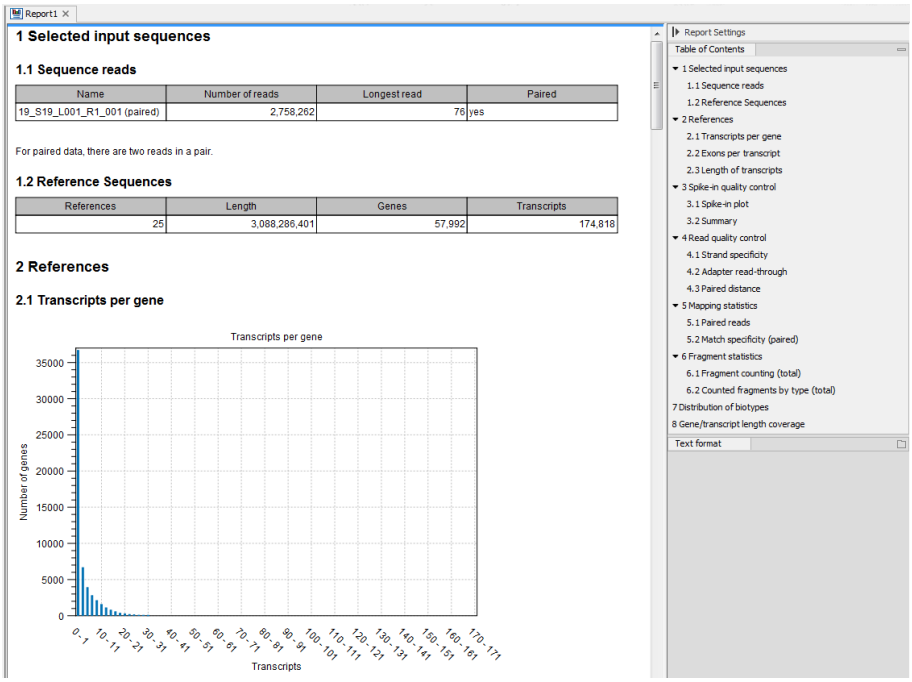


Figure 29.16: *Report of an RNA-Seq run.*

and their lengths.

References Information about the total number of genes and transcripts found in the reference:

- **Transcripts per gene**. A graph showing the number of transcripts per gene.
- **Exons per transcript**. A graph showing the number of exons per transcript.
- **Length of transcripts**. A graph showing the distribution of transcript lengths.

Spike-in quality control

- **Spike-in plot**. A plot shows the expression of each spike-in as a function of the known concentration of that spike-in (see figure 29.17 to see an optimal spike-in plot).
- **Summary table**. A table provides more details on the spike-in detection. Figure 29.18 shows a failed spike-in control, with a table where results that require attention are highlighted in pink.

  Under the table, a **warning message** explains what the optimal value was, and offers some troubleshooting measures: When samples have poor correlation $(R^2 < 0.8)$ between known and measured spike-in concentrations, it indicates problems with the spike-in protocol, or a more serious problem with the sample. To troubleshoot, check that the correct spike-in file has been selected, and control the integrity of the sample RNA. Also, if fewer than 10000 reads mapped to spike-ins, check that the correct spike-in sequences are specified, and consider using more spike-in mix in future experiments.

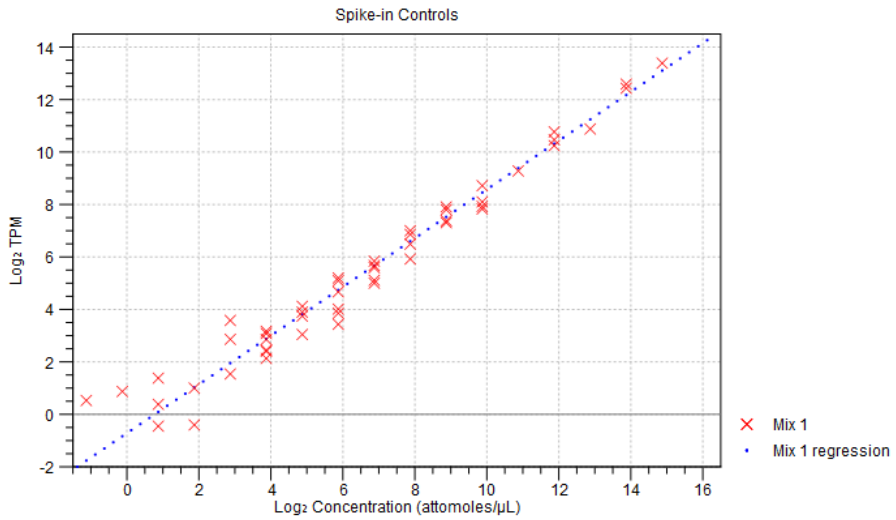
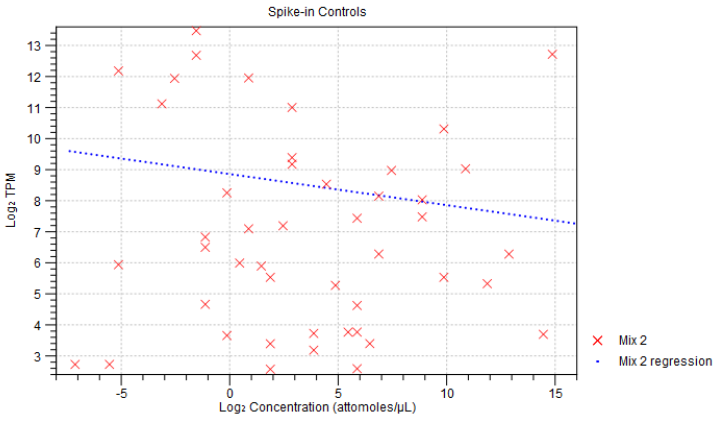


Figure 29.17: *Spike-in plot showing how the points fall close to the regression line at high concentration.*



3.2 Summary

Number of spike-ins detected	46/92
R^2	0.05
Reads mapped to spike-ins	7,336
% of reads mapped to spike-ins	3.10
Lower limit of detection (attomoles/µL)	0.03

Figure 29.18: *Summary table where less than optimal results are highlighted.*

Read quality control This section includes:

- A **strand specificity table** that indicates the direction of the RNA fragment that generated the read. In a strand-specific protocol almost all reads are generated from a specific orientation but otherwise, a mix of both orientations is expected.
  - A warning message will appear if over 90% of reads were mapped in the same orientation but the tool was run without using a strand specific setting ("Forward"/"Reverse").
  - If over 25% of the reads were filtered away due to the strand specific setting, try to re-run the tool with strand specific setting "Both". However, if a strand-specific protocol was used, library preparation may have failed.
- A **percentage of mapped paired-end reads containing read-through adapters**. If present

in above 10% of the reads, adapters may lead to false positive variant calls or incorrect transcript quantification (because reads must align within transcript annotations to be counted towards expression). Read-through adapters can be removed using the Trim Reads tool. Note that single base extensions such as TA overhangs will also be classed as read-through adapters, and in these cases the additional base should also be trimmed. In future experiments, consider selecting fragments that are longer than the read size.

- A **paired distance graph** (only included if paired reads are used) shows the distribution of paired-end distances, which is equivalent to the distribution of sequenced RNA fragment sizes. There should be a single broad peak at the target fragment size. An asymmetric peak may indicate problems in size selection.

Mapping statistics Shows statistics on:

- **Paired reads** or **Single reads**. The table included depends on the reads used. The table shows the number of reads mapped or unmapped, and in the case of paired reads, how many reads mapped in pairs and in broken pairs.

  If over 50% of the reads did not map, and the correct reference genome was selected, this indicates a serious problem with the sample. To troubleshoot, the report offers the following options:

  - Check that the correct reference genome and any relevant gene/mRNA tracks have been provided.
  - The mapping parameters may be too strict. Try resetting them to the default values.
  - Try mapping the un-mapped reads against possible contaminants. If the sample is contaminated, enrich for the target species before library preparation in future experiments.
  - Library preparation may have failed. Check the quality of the sample RNA.

  In case paired reads are used and over 40% of them mapped as broken pairs, the report hints that there could be problems with the tool settings, a low quality reference sequence, or incomplete gene/mRNA annotations. It could also indicate a more serious problem with the sample. To troubleshoot, it is suggested to:

  - Check that the correct reference genome and any relevant gene/mRNA tracks have been provided.
  - Try re-running the tool with the "Auto-detect paired distances" option selected.
  - Check that the paired-end distances on the reads are set correctly. These are shown in the "Element Information" view on the reads. If these are correct, try re-running the tool without the "Auto-detect paired distances" option.
  - Try mapping the reads against possible contaminants. If the sample is contaminated, enrich for the target species before library preparation in future experiments.

- **Match specificity**. Shows a graph of the number of match positions for the reads. Most reads will be mapped 0 or 1 time, but there will also be reads matching more than once in the reference. The maximum number of match positions is limited in the **Maximum number of hits for a read** setting in figure 29.4. Note that the number of reads that are mapped 0 times includes both the number of reads that cannot be mapped at all and the number of reads that matches to more than the **Maximum number of hits for a read** parameter.

Fragment statistics

- **Fragment counting**. Lists the total number of fragments used for calculating expression, divided into uniquely and non-specifically mapped reads, as well as uncounted fragments (see the point below on match specificity for details).
- **Counted fragments by type**. Divides the fragments that are counted into different types, e.g., uniquely mapped, non-specifically mapped, mapped. A last column gives the percentage of fragments mapped for a particular type.
  - **Total gene reads**. All reads that map to the gene.
  - **- Intron**. From the total gene reads, reads that fall partly or entirely within an intron.
  - **- Exon**. From the total gene reads, reads that fall entirely within an exon or in an exon-exon junction.
  - **- - Exon**. From the total gene - exon reads, reads that map completely within an exon
  - **- - Exon-exon**. From the total gene - exon reads, reads that map across an exon junction as specified in figure 29.13.
  - **Intergenic**. All reads that map partly or entirely between genes.
  - **Total**. Total amount of reads for a particular type.

Distribution of biotypes Table generated from biotype annotations present on the input gene or mRNA tracks. If using both gene and mRNA tracks, the biotypes in the report are taken from the mRNA track.

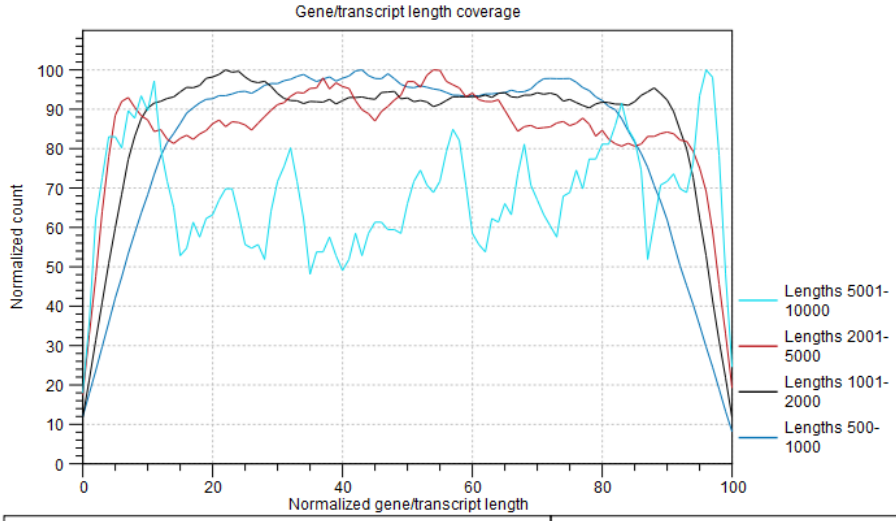
- For genes, biotypes can be any of the following columns: "gene_biotype", "biotype", "gbkey", "type". The first one in this list is chosen.
- For transcripts, biotypes can be any of the following columns: "transcript_biotype", "biotype", "gbkey", "type". The first one in this list is chosen.

The biotypes are "as a percentage of all transcripts" or "as a percentage of all genes". For a poly-A enrichment experiment, it is expected that the majority of reads correspond to protein-coding regions. For an rRNA depletion protocol, a variety of non-coding RNA regions may also be observed. The percentage of reads mapping to rRNA should usually be <15%.

If over 15% of the reads mapped to rRNA, it could be that the poly-A enrichment/rRNA depletion protocol failed. The sample can still be used for differential expression and variant calling, but expression values such as TPM and RPKM may not be comparable to those of other samples. To troubleshoot the issues in future experiments, check for rRNA depletion prior to library preparation. Also, if an rRNA depletion kit was used, check that the kit matches the species being studied.

Gene/transcript length coverage Plot showing the normalized coverage across a gene/transcript body for four different groupings of gene/transcript length (figure 29.19).

The lines should be flat in the center of the plot, and the plot should be approximately symmetric. An erratic line may indicate that there are few genes/transcripts in the given length range. Lines



Difference between average 3' and 5' normalized counts	-2.26
% reads mapping to genes/transcripts > 10000 bp	0.06

The plot shows the normalized coverage across a gene/transcript body for four different groupings of gene/transcript length. The lines should be flat in the center of the plot, and the plot should be approximately symmetric. An erratic line may indicate that there are few genes/transcripts in the given length range.

Figure 29.19: *Gene/transcript length coverage plot.*

showing normalized count higher on the 3'end indicates the presence of polyA tails in the reads, consequence of degraded RNAs. Future experiments may benefit from using an rRNA depletion protocol.

In the table below the plot, a difference between average 3' and 5' normalized counts higher than 25 warns that variants may not be called in low coverage regions, and that TPM or RPKM values may be unreliable. Most transcripts are <10000 bp long, so a warning is raised if many reads map to features longer than this. One possible cause is that no mRNA track has been provided for an organism with extensive splicing.

29.1.9 Gene fusion reporting

When using paired data, there is also an option to create an annotation track summarizing the evidence for gene fusions. An example is shown in figure 29.20.

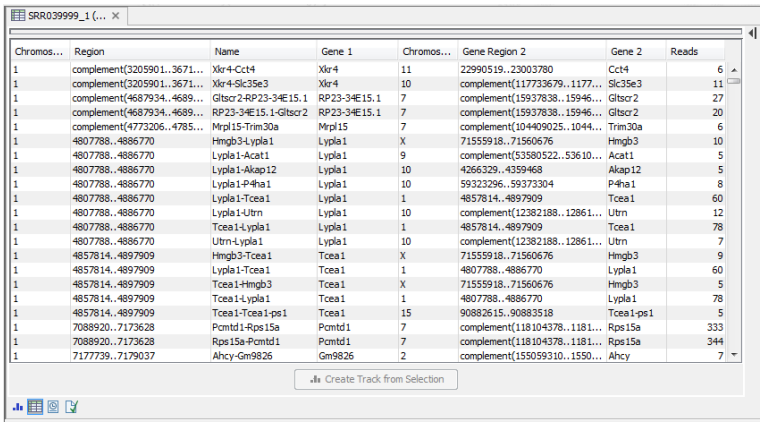


Figure 29.20: *An example of a gene fusion table.*

Each row represents one gene where read pairs suggest it could be fused with another gene. This means that each fusion is represented by two rows.

The **Minimum read count** option in figure 29.11 is used to make sure that only combinations of genes supported by at least this number of read pairs are included. The default value is 5, which means that at least 5 pairs need to connect two genes in order to report it in the result.

The result table shows the following information for each row:

- **Name**. The name of the fusion (the two gene names combined).
- **Information per gene**. Gene name, chromosome and position are included for both genes.
- **Reads**. How many reads that are mapped across the two genes.

Note that the reporting of gene fusions is very simple and should be analyzed in much greater detail before any evidence of gene fusions can be verified. The table should be considered more of a pointer to genes to explore rather than evidence of gene fusions. Please note that you can include the fusion genes track in a track list together with the reads tracks to investigate the mapping patterns in greater detail:

File | **New** | **Genome Browser View (**  **)**

29.2 Create Combined RNA-Seq Report

With the Create Combined RNA-Seq Report tool, you can generate an overview of several RNA-seq experiments by combining in one document several RNA-Seq reports. The description of the various sections included in the report, as well as the cutoff values that trigger warnings for sub-optimal data are the same as described for RNA-Seq reports (see section 29.1.8). The combined report can be exported in PDF or Excel format.

To start the tool:

Toolbox | **RNA-Seq Analysis (**  **)** | **Create Combined RNA-Seq Report**

In the wizard that opens (figure 29.21), select several RNA-seq reports and click **Next**. Note that while it seems possible to select any kind of reports in that dialog, only RNA-seq reports generated by *Biomedical Genomics Workbench 5.0.1*or higher will be compiled in the combined RNA-seq report.

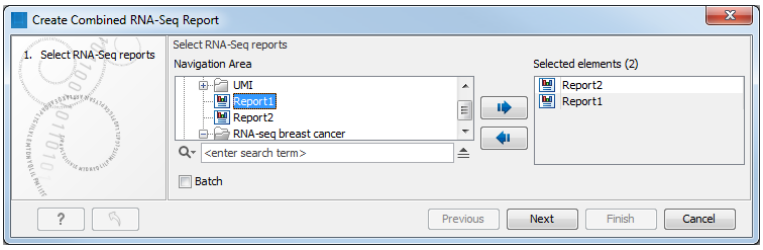


Figure 29.21: *The Create Combined RNA-Seq Report tool.*

In the Result handling window, choose whether you want to open or save the combined report. When saving, specify where you would like to save the combined report in the Navigation Area, and click **Finish**.

The results of the individual reports are compiled in tables, and organized in the same sections than the ones were present in the individual input reports. This layout enables a quick overview of a series of experiments as well as the different troubleshooting options for dealing with sub-optimal data quality (see figure 29.22 for an example of the first four sections of a combined report).

1 Read count statistics

Sample name	Read count	Single, mapped %	Single, unmapped %	Paired, mapped pairs %	Paired, broken pairs %	Paired, not mapped %
Report2	1,000,000	N/A	N/A	39.98	3.69	N/A
Report1	2,758,262	N/A	N/A	93.21	4.13	N/A

For paired data, there are two reads in a pair.

2 Fragment counting statistics

Sample name	Mapped to genes %	Mapped to intergenic %
Report2	67.50	32.50
Report1	98.51	1.49

Default counting scheme ('Fragment counts'): An intact pair is counted as one, broken pairs are ignored.

3 Spike-in quality control

Sample name	Detected spike-ins	R^2	Reads mapped to spike-ins	% of reads mapped to spike-ins	Lower limit of detection (attomoles/µL)
Report2	46/92	0.05	7,336	3.10	0.03
Report1	54/92	0.97	24,667	1.76	7.32

R^2: The sample has poor correlation ($R^2 < 0.8$) between known and measured spike-in concentrations. This may indicate problems with the spike-in protocol, or a more serious problem with the sample.
- Check that the correct spike-in file has been selected.
- Check the integrity of the sample RNA.

Reads mapped to spike-ins: Fewer than 10000 reads mapped to spike-ins.
- Check that the correct spike-in sequences are specified.
- Consider using more spike-in mix in future experiments.

4 Strand specificity

Sample name	Strand specific setting	Forward % of reads mapped	Reverse % of reads mapped	Ignored reads (wrong strand)	Ignored reads % (wrong strand)
Report2	Both	50.50	49.50	0	0.00
Report1	Both	96.01	3.99	1,644	0.06

Strand specific setting: >90% of reads were mapped in the same orientation. Consider re-running the tool with a strand specific setting ("Forward"/" Reverse").

Figure 29.22: *An example combined report.*

29.3 Create fold change track

The **Create Fold Change Track** tool can be used to compare RNA expression levels in two samples (e.g., matched tumor and normal samples).

After RNA-seq analysis, the resulting expression tracks can be compared using the **Create Fold Change Track** tool. For each gene or transcript, this tool will compute the ratio of the expression value in the case track to the expression value of the same gene in the control track. The tool enables filtering on the basis of fold-changes as well as expression values. This can give a quick first indication of possible differentially expressed genes or transcripts between two RNA-seq samples. For a more detailed and statistically robust analysis, we recommend running a Statistical Analysis tool in the *Biomedical Genomics Workbench*.

The tool for creating a fold-change track can be found here:

Toolbox | **RNA-Seq Analysis** ( ) | **Create Fold Change Track** ( )

After starting the Create Fold Change Track tool, select the expression track corresponding to the case sample in the input dialog, as shown in figure 29.23, and click on the button labeled **Next**.

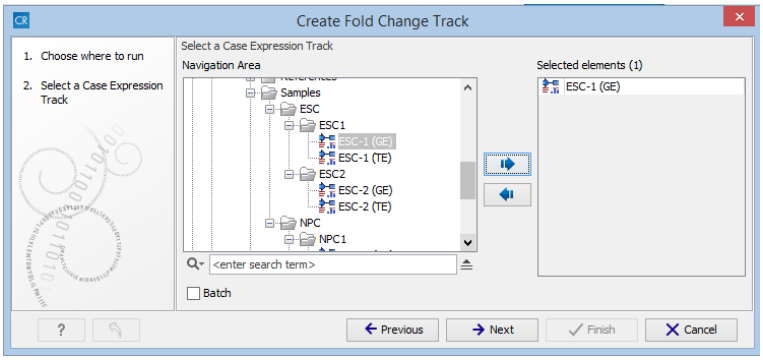


Figure 29.23: *Choosing the case expression track for the Create Fold Change Track tool.*

Setting the parameters

You are now presented with the wizard step shown in figure 29.24 where you can set the following input parameters:

- **Control expression track** At the top of the wizard, a control expression track must be chosen. The control expression track will be used as the reference in creating the fold-change track.
- **Log fold change value** In the middle of the wizard, the parameters relating to the reported fold-change values can be specified.
  - *Scale* Determines on what scale the fold-changes should be reported and interpreted:
    - **Raw** The fold-changes will be reported as a direct ratio: $\text{case expression} = \frac{\text{case expression}}{\text{control expression}}$
    - **Log-2** The fold-changes will be reported as log-2 values: $\text{case expression} = \log_2\left(\frac{\text{case expression}}{\text{control expression}}\right)$
    - **Log-10** The fold-changes will be reported as log-10 values: $\text{case expression} = \log_{10}\left(\frac{\text{case expression}}{\text{control expression}}\right)$

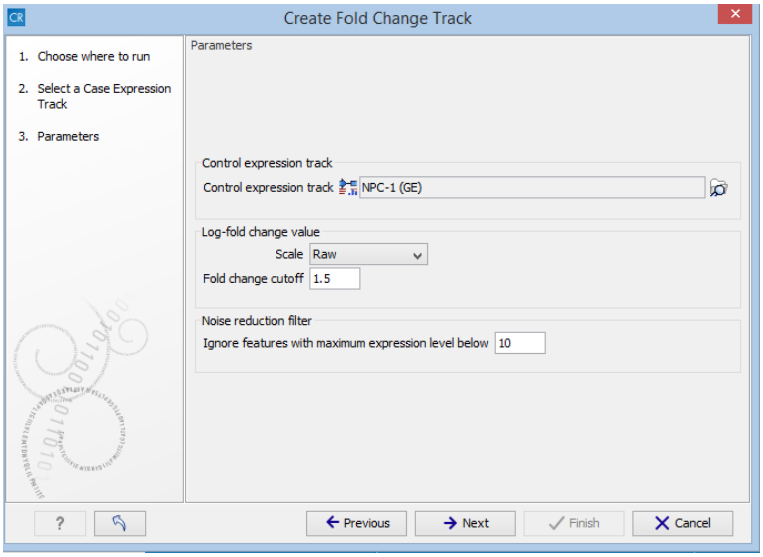


Figure 29.24: *Setting the input parameters for the Create Fold Change Track tool.*

  - ∗ **Natural logarithm** The fold-changes will be reported as log-e values: case expression $= \log\left(\frac{\text{case expression}}{\text{control expression}}\right)$
- *Fold-change cutoff* Enables filtering in the results based on fold-changes. Note that the value entered into the fold-change cutoff field will be interpreted according to the scale specified in the *scale* parameter. For example, if the expression level of a gene is 100 in the case sample and 200 in a control sample, then a cutoff of 1.5 with the scale parameter set to **Raw** will not result in the gene being filtered on the basis of fold-changes, and the gene will appear in the final results. However, if the scale parameter is set to **Log10**, the gene will be filtered and will not appear in the final results. To include all genes or transcripts in the output without filtering on the basis of fold-changes, enter 0 in this field (this will work for any value specified for the *scale* parameter).

- **Noise reduction filter** A second level of filtering is possible based on absolute expression levels. The rationale behind this filtering is that seemingly very large fold-changes can occur by chance if the expression levels are very low in both samples, creating a false-positive noise in the resulting fold-change track. Therefore, it is desirable to require a certain level of expression for a gene in at least one of the samples. This can be specified using the *Ignore features with maximum expression level below* parameter.
  - *Ignore features with maximum expression level below* The value entered in this field corresponds to the minimum expression level required in *either* the case *or* the control track, in order for the gene or transcript to appear in the results. If the expression level does not exceed this value in either the case or the control sample, the gene or transcript will be filtered out from the final results. Entering a value of 0 for this parameter will not filter any genes or transcripts based on absolute expression levels.

When finished with setting the parameters, click **Next**.

Interpreting the results

The Create Fold Change Track tool will produce an annotation track, including the genes or transcripts that were not filtered out based on the filtering criteria. Each feature in this track is annotated with the following information:

- **Fold-change** Represents the fold-change according to the specified scale. A positive value indicates that the expression level was higher in the case. A negative value indicates that the expression level was higher in the control. If **Raw** was specified for the *scale* parameter, a value of 0 represents no difference. If a logarithm-based value was specified for the *scale* parameter, a value of 1 represents no difference.
- **Difference** Represents the difference in expressions. A positive value indicates that the expression level was higher in the case. A negative value indicates the expression level was higher in the control.
- **Maximum expression** Represents the larger of the expression values observed in the case and the control samples.
- **Expression (case)** Gives the expression value in the case sample
- **Expression (control)** Gives the expression value in the control sample

As is the case with all annotation tracks in the *Biomedical Genomics Workbench*, it is possible to sort and filter the results based on any of the above values, and to create track lists for further analysis of the results.

29.4 PCA for RNA-Seq

Principal Component Analysis makes it possible to project a high-dimensional dataset (where the number of dimensions equals the number of genes or transcripts) onto two or three dimensions. This helps in identifying outlying samples for quality control, and gives a feeling for the principal causes of variation in a dataset. The analysis proceeds by transforming a large set of variables (in this case, the counts for each individual gene or transcript) to a smaller set of orthogonal principal components. The first principal component specifies the direction with the largest variability in the data, the second component is the direction with the second largest variation, and so on.

The **PCA for RNA-Seq** tool clusters samples in 2D or 3D. Known metadata about each sample is added as an overlay. To start the analysis:

Toolbox | **RNA-Seq Analysis** | **PCA for RNA-Seq ()**

Select a number of expression tracks () and click **Next**.

29.4.1 Principal component analysis plot (2D)

The default view is a two-dimensional principal component plot as shown in figure 29.25.

The plot shows the projection of the samples onto the two-dimensional space spanned by the first and second principal components of the covariance matrix. The expression levels used as input are normalized log CPM values, see section 29.

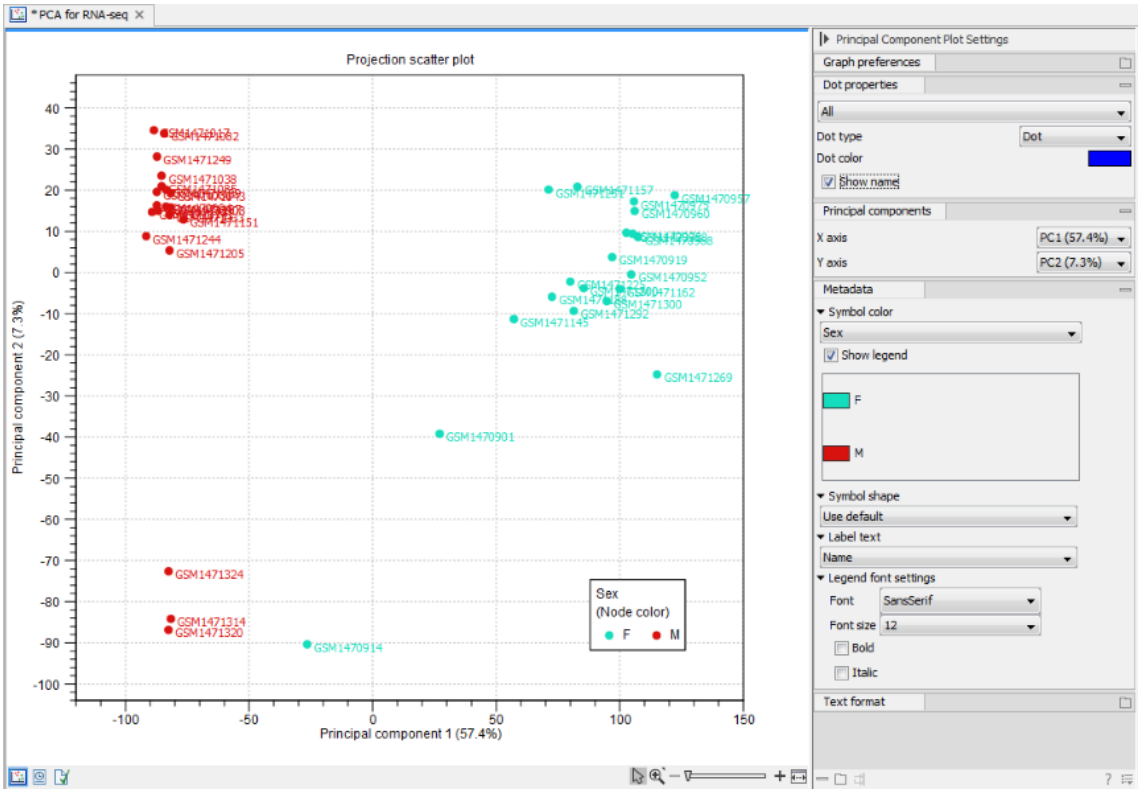


Figure 29.25: *A principal component plot.*

The view settings can be adjusted using the **Side Panel**. 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.
- **Tick type** Determines 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 with options for adjusting the line appearance.

Below the general preferences, you find the **Dot properties**:

- **Drop down menu** In this you select the expression tracks to which following choices apply.
- **Dot type** Allows you to choose between different dot types.

- **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 Dot properties may be overridden when the Metadata options are used to control the visual appearance (see below).

The **Principal Components** group determines which two principal components are used in the 2D plot. By default, the first principal component is shown for the X axis and the second principal component is shown for the Y axis. The value after the principal component identifier (for example "PC1 (72.5 %)") displays the amount of variance explained by this particular principal component.

The **Metadata** group allows metadata associated with the Expression tracks to be visualized in a number of ways:

- **Symbol color** Colors are assigned based on a categorical factor in the metadata table.
- **Symbol shape** Shape is assigned based on a categorical factor in the metadata table.
- **Label text** Dots are labeled according to the values in a given metadata column.
- **Legend font settings** contains options to adjust the display of labels.

The graph and 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).

29.4.2 Principal component analysis plot (3D)

The principal component plot may also be displayed in 3D. The 3D view is accessible through the view buttons at the bottom of the panel.

Notice that the 3D PCA rendering feature requires a graphics card capable of supporting OpenGL 2.0. Please make sure the latest driver for the graphics card is installed. Indirect rendering (such as x11 forwarding through ssh), remote desktop connection/VNC, and running in virtual machines is not supported.

The 3D view may be rotated by dragging on the view with the left mouse button pressed. It is possible to pan the view by dragging with the right mouse button pressed. Zooming can be done either using the mouse scroll wheel, or by dragging with both left and right mouse button pressed. It is also possible to center and zoom to a sample simply by clicking on it. Clicking outside any sample (or clicking with the right mouse button) restores the zoom and centering.

The **Side Panel** offers a number of options to change the appearance of the 3D principal component plot:

The **View settings** group makes it possible to toggle the coordinate system on and off, and adjust the text and background color. It is also possible to enable **Fog**, which dims distant objects in order to improve the depth perception.

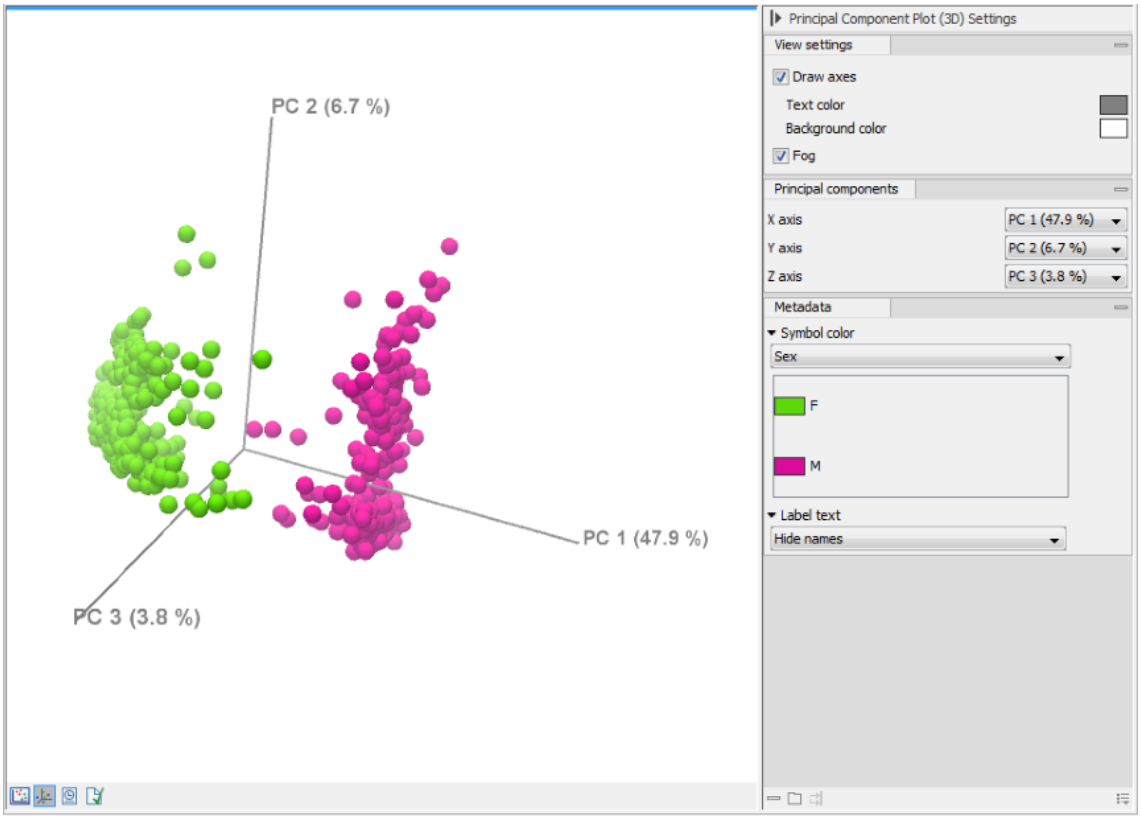


Figure 29.26: *A principal component 3D plot.*

The **Principal Components** group determines which principal components are used in the 3D plot. The value after the principal component identifier (for example "PC 1 (72.5 %)") displays the amount of variance explained by this particular principal component.

The **Metadata** group allows metadata associated with the Expression tracks to be visualized using color or as text:

- **Symbol color** Colors are assigned based on a categorical factor in the metadata table.
- **Label text** Samples are labeled according to the values in a given metadata column. If 'Show names' is selected, the samples will be labeled according to their name (as shown in the **Navigation Area**).

To save the current view as an image, press the **Graphics** button in the Workbench toolbar. Next, select the location where you wish to save the image, select file format (PNG, JPEG, or TIFF), and provide a name, if you wish to use another name than the default name.

It is possible to save the current view settings (including camera settings) using the **Side Panel** view settings options, see section 4.6.

29.5 Differential Expression for RNA-Seq

The **Differential Expression for RNA-Seq** tool performs a statistical differential expression test for a set of Expression Tracks. It uses multi-factorial statistics based on a negative binomial

GLM. The tool supports paired designs and can control for batch effects. The statistical analysis is described in more detail in section 29.5.1.

To run the Differential Expression for RNA-Seq analysis:

Toolbox | **RNA-Seq Analysis** | **Differential Expression for RNA-Seq**

Select a number of Expression tracks ( ) and click **Next**. For Expression Tracks (TE), the values used as input are "Total transcript reads". For Gene Expression Tracks (GE), the values used depend on whether an eukaryotic or prokaryotic organism is analyzed, i.e. if the option "Genome annotated with Genes and transcripts" or "Genome annotated with Genes only" is used. For Eukaryotes the values are "Total Exon Reads", whereas for Prokaryotes the values are "Total Gene Reads".

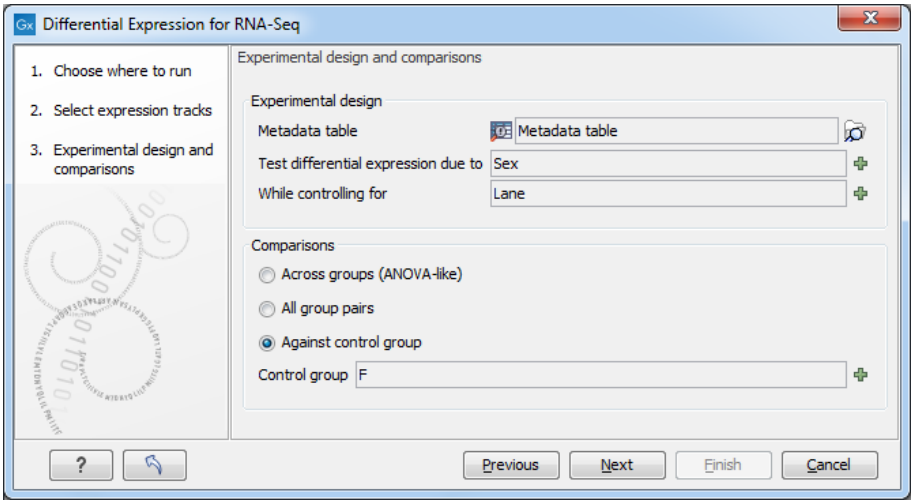


Figure 29.27: *Setting up the experimental design and comparisons.*

This will display the wizard shown in figure 29.27.

In the **Experimental design** panel, a Metadata table must be selected that describes the factors and groups for all the samples.

- **Metadata table** The metadata table describing the factors for the selected Expression tracks.
- **Test differential expression due to** Specify the one factor differential expression is tested for.
- **While controlling for** Specify confounding factors, i.e., factors that are not of primary interest, but may affect gene expression.

The **Comparisons** panel determines the number and type of statistical comparison tracks output by the tool (see section 29.5.2 for more details).

How many replicates do I need? The Differential Expression for RNA-Seq tool is capable of running without replicates, but this is not recommended and the results should be treated with caution. In general it is desirable to have as many biological replicates as possible – typically at least 3. Replication is important in that it allows the 'within group' variation to be accurately estimated for a gene. In the absence of replication, the Differential Expression for RNA-Seq tool assumes that genes with similar average expression levels have similar variability.

Technical or biological replicates? [Auer and Doerge, 2010] illustrates the importance of *biological replicates* with the example of an alien visiting Earth. The alien wishes to know if men are taller than women. It abducts one man and one woman, and measures their heights several times i.e. performs several *technical replicates*. However, in the absence of *biological replicates*, the alien would erroneously conclude that women are taller than men if this was the case in the two abducted individuals.

29.5.1 The statistical model

Each gene is modeled by a separate Generalized Linear Model (GLM). The use of the GLM formalism allows us to fit curves to expression values without assuming that the error on the values is normally distributed. Similarly to EdgeR and DESeq, we assume that the read counts follow a Negative Binomial distribution.

The Negative Binomial distribution can be understood as a 'Gamma-Poisson' mixture distribution i.e. the distribution resulting from a mixture of Poisson distributions, where the Poisson parameter λ is itself Gamma-distributed. In an RNA-Seq context, this Gamma distribution is controlled by the **dispersion** parameter, such that the Negative Binomial distribution reduces to a Poisson distribution when the dispersion is zero.

Fitting a GLM to expression data

It is easiest to understand how the GLM model works through an example. Imagine an experiment looking at the effect of two drug treatments while controlling for the gender of a patient:

- **Test differential expression due to** Treatment with three groups: drugA, drugB, placebo
- **While controlling for** Gender with groups: Male, Female

In an abuse of mathematical notation, the underlying GLM for each gene looks like

$$\log y_i = (\text{placebo and Male}) + \text{drugA} + \text{drugB} + \text{Female} + \text{constant}_\text{i} \tag{29.1}$$

where y_i is the expression level for the gene in sample i; the combined term $(\text{placebo and Male})$ describes an arbitrarily chosen baseline expression level (of males being given a placebo); and the other terms drugA, drugB and Female are numbers describing the effect of each group with respect to this baseline. The constant_i accounts for differences in the library size between samples. For example, if a patient is male and given a placebo we predict the expression level to be

$$\log y_i = (\text{placebo and Male}) + \text{constant}_\text{i}.$$

If instead he had been given drug B, we would predict the expression level y_i to be augmented with the drugB coefficient, resulting in

$$\log y_i = (\text{placebo and Male}) + \text{drugB} + \text{constant}_\text{i}.$$

We assume that the expression levels y_i follow a Negative Binomial distribution. This distribution has a free parameter, the dispersion. The greater the dispersion, the greater the variation in expression levels for a gene.

The most likely values of the dispersion and coefficients, drugA, drugB and Female, are determined simultaneously by fitting the GLM to the data. To see why this simultaneous fitting is necessary, imagine an experiment where we observe counts {3,10,4} for Males and {30,20,8} for Females. The most natural fit is for the coefficient Female to have a two-fold change and for the dispersion to be small, but an alternative fit has no fold change and a larger dispersion. Under this second fit the variation in the counts is greater, and it is just by chance that all three Female values are larger than all three Male values.

Refining the estimate of dispersion

Much research has gone into refining the dispersion estimates of GLM fits. One important observation is that the GLM dispersion for a gene is often too low, because it is a *sample* dispersion rather than a *population* dispersion. We correct for this using the Cox-Reid adjusted likelihood, as in the multi-factorial EdgeR method [Robinson et al., 2010]. [2]

A second observation that can be used to improve the dispersion estimate, is that genes with the same average expression often have similar dispersions. To make use of this observation, we follow [Robinson et al., 2010] in estimating genewise dispersions from a linear combination of the likelihood for the gene of interest and neighboring genes with similar average expression levels. The weighting in this combination depends on the number of samples in an experiment, such that the neighbors have most weight when there are no replicates, and little effect when the number of replicates is high.

Statistical testing

The final GLM fit and dispersion estimate allows us to calculate the total likelihood of the model given the data, and the uncertainty on each fitted coefficient. The two statistical tests each make use of one of these values.

Wald test Tests whether a given coefficient is non-zero. This test is used in the **All group pairs** and **Against control group** comparisons. For example, to test whether there is a difference between patients treated with a placebo, and those treated with drugB, we would use the Wald test to determine if the drugB coefficient is non-zero.

[2] To understand the purpose of the correction, it may help to consider the analogous situation of calculation of the variance of normally distributed measurements. One approach would be to calculate $\frac{1}{n}\sum(x_i - \overline{x})^2$, but this is the *sample* variance and often too low. A commonly used correction for the *population* variance is: $\frac{1}{n-1}\sum(x_i - \overline{x})^2$.

Likelihood Ratio test Fits two GLMs, one with the given coefficients and one without. The more important the coefficients are, the greater the ratio of the likelihoods of the two models. This test is used in the Across groups (ANOVA-like) comparison. If we wanted to test whether either drug had an effect, we would compare the likelihoods of the GLM described in equation 29.1 with those in the reduced GLM $\log y_i = (\mathrm{Male}) + \mathrm{Female} + \mathrm{constant_i}$.

29.5.2 Output of the Differential Expression for RNA-Seq tool

The Differential Expression for RNA-Seq tool produces different numbers and types of statistical comparison tracks depending on the settings of the **Comparisons** panel. Depending on the choice either a Wald test or a Likelihood Ratio test is used. For example, assume that we test a factor called 'Tissue' with three groups: skin, liver, brain.

- **Across groups (ANOVA-like)** This mode tests for the effect of a factor across all groups.
  - *Outputs produced*: "Due to Tissue"
  - *Test used*: Likelihood ratio test
  - *Fold change* reports: The maximum pairwise fold change between any two of the three tissue types.
  - *Max of group means* reports: The maximum of the average group RPKM values among any of the tissue types for a gene.
- **All group pairs** tests for differences between all pairs of groups in a factor.
  - *Outputs produced*: "skin vs. liver", "skin vs. brain", "liver vs. brain"
  - *Test used*: Wald test
  - *Fold change* reports: The fold change in the defined order between the named pair of tissue types.
  - *Max of group means* reports: The maximum of the average group RPKM values between the two named tissue types.
- **Against control group** This mode tests for differences between all the groups in a factor and the named reference group. In this example the reference group is skin.
  - *Outputs produced*: "liver vs. skin", "brain vs. skin"
  - *Test used*: Wald test
  - *Fold change* reports: The fold change in the defined order between the named pair of tissue types.
  - *Max of group means* reports: The maximum of the average group RPKM values between the two named tissue types.

Note: Fold changes are calculated from the GLM, which corrects for differences in library size between the samples and the effects of confounding factors. It is therefore not possible to derive these fold changes from the original counts by simple algebraic calculations.

29.5.3 Statistical comparison tracks

The Differential Expression for RNA-Seq tool will output one or more statistical comparison tracks.

An example of a statistical comparison track is shown in figure 29.28. Statistical comparison tracks make it possible to show differential expression data alongside other kinds of tracks in a genomic context.

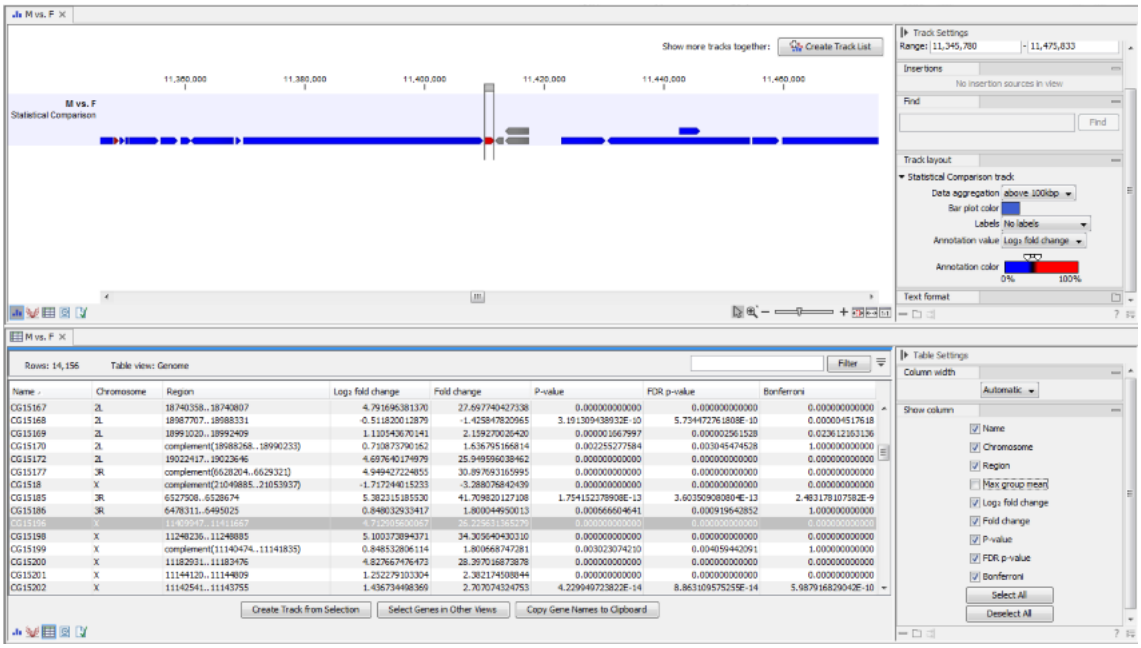


Figure 29.28: *Statistical comparison track view.*

In particular, the Fold Change value will tell you how expression levels in group 2 are relative to that in group 1.

- If expression values in group 2 are twice as large as in group 1, the fold change will be +2.
- If expression values in group 1 are twice as large as in group 2, the fold change will be -2.

Note that it is not possible to derive these fold changes from the CPM values by simple algebraic calculations as the Differential Expression for RNA-Seq tool works by fitting a statistical model (which accounts for differences in sequencing-depth) to raw counts. Keep also in mind that p-values smaller than 10e-16 will be reported as equal to zero, and as a consequence these values will not appear on the volcano plot.

The track layout of the statistical comparison track can be customized as follows:

- **Data aggregation** Allows you to specify whether the information in the track should be shown in detail or whether you wish to aggregate data. By aggregating data you decrease the detail level, but increase the speed of the data display process, which is of particular interest when working with big data sets. The threshold (in bp) for when data should be aggregated can be specified with the drop-down box. The threshold describes the unit (or "bucket") size in base pairs, above which the data will start being aggregated. The bucket size depends on the track length and the zoom level. Hence, a data aggregation threshold with a low value will only show details when zoomed in, whereas a high value means that

you can see details even when zoomed out. Please note that when using the high values, it will take longer time to display the data on the screen.

- **Bar plot color** Selects the color of aggregated data.
- **Labels** Determines where the gene name should be shown.
- **Annotation value** The value that is graphically shown in detail view:
  - **Max group means** For each group in the statistical comparison, the average RPKM is calculated. This value is the maximum of the average RPKM's.
  - **$\mathbf{Log_2}$ fold change** The logarithmic fold change.
  - **Fold change** The (signed) fold change. Genes/transcripts that are not observed in any sample have undefined fold changes and are reported as NaN (not a number).
  - **P-value** Standard p-value. Genes/transcripts that are not observed in any sample have undefined p-values and are reported as NaN (not a number).
  - **FDR p-value** The false discovery rate corrected p-value.
  - **Bonferroni** The Bonferroni corrected p-value.
- **Annotation color** Determines how the annotation value is mapped onto a color.

The expression track table view has three button.

- The "Create track from Selection" will create a Track using selected rows.
- The "Select Genes in Other Views" button finds and selects the currently selected genes and transcripts in all other open expression track table views.
- The "Copy Gene Names to Clipboard" button copies the currently selected gene names to the clipboard.

29.5.4 The volcano plot

Statistical comparison tracks also offer a volcano plot view.

An example of a volcano plot is shown in figure 30.70.

The volcano plot shows the relationship between the p-values of a statistical test and the fold changes among the samples. The $\log_2$ fold changes are plotted on the x-axis, and the $-\log_{10}$ p-values are plotted on the y-axis. Features of interest are typically those in the upper left and right hand corners of the volcano plot, as these have large fold changes (lie far from $x = 0$) and are statistically significant (have large y-values). Please note that p-values smaller than 10e-16 are reported as equal to zero, and as a consequence cannot appear on the volcano plot.

It is possible to change the type of p-value from the side panel (see below).

The view settings can be adjusted using the **Side Panel**. 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.

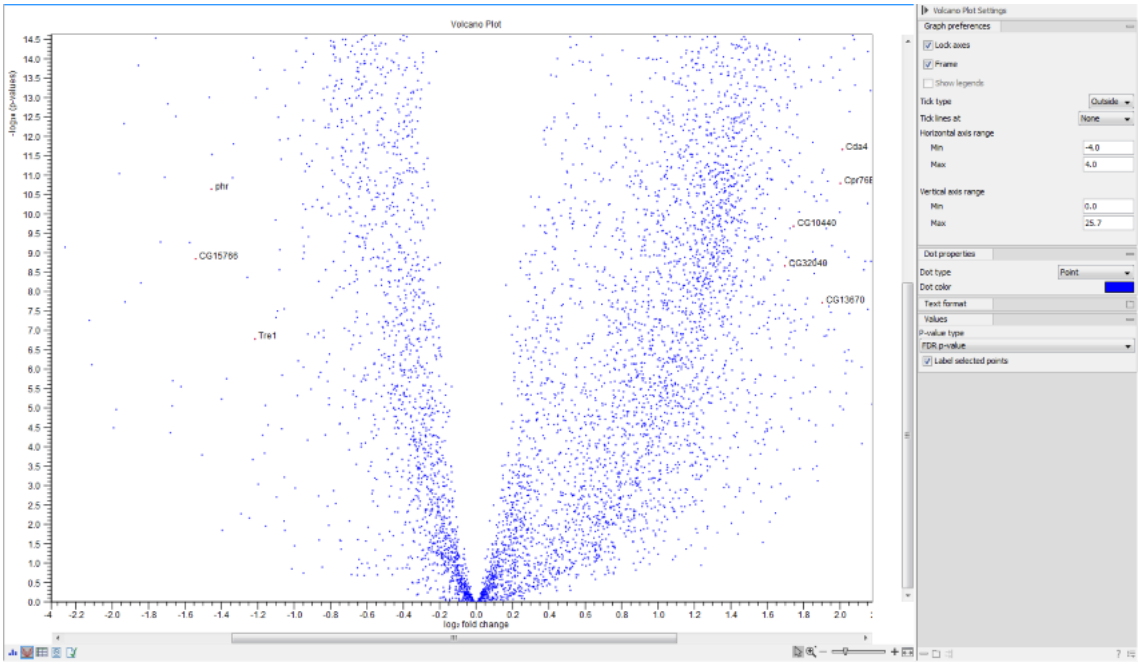


Figure 29.29: *Volcano plot.*

- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

Below the general preferences, you find the **Dot properties** and **Text format**, where you can adjust the coloring and appearance of the dots and text.

At the bottom are options for choosing which values to display:

- **P-value type** Selects which type of p-value to use.
- **Label selected points** Chooses whether selected points should be labeled.

Note that if you wish to use the same settings next time you open a volcano plot, you need to save the settings of the **Side Panel** .

29.6 Create Heat Map for RNA-Seq

The **Create Heat Map** tool simultaneously clusters samples and features, showing a two dimensional heat map of expression values. Each column corresponds to one sample, and each row corresponds to a feature (a gene or a transcript). The samples and features are both hierarchically clustered. Known metadata about each sample is added as an overlay.

The Create Heat Map for RNA-Seq tool uses the TMM normalization (described in section 29) to make samples comparable, then does a z-score normalization to make features comparable.

29.6.1 Clustering of features and samples

The hierarchical clustering clusters features by the similarity of their expression profiles over the set of samples. It clusters samples by the similarity of expression patterns over their features.

Each clustering has a tree structure that is generated by

1. Letting each feature or 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 contains all the features or samples).

The tree is drawn so that the distances between clusters are reflected by the lengths of the branches in the tree.

To create a heat map:

Toolbox | **RNA-Seq Analysis** | **Create Heat Map for RNA-Seq (****)**

Select at least two expression tracks () and click **Next**.

This will display the wizard shown in figure 29.30. 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 or samples should be calculated. The cluster linkage specifies how the distance between two clusters, each consisting of a number of features or samples, should 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

$$|u - v| = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}.$$

- **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

$$|u - v| = \sum_{i=1}^{n} |u_i - v_i|.$$

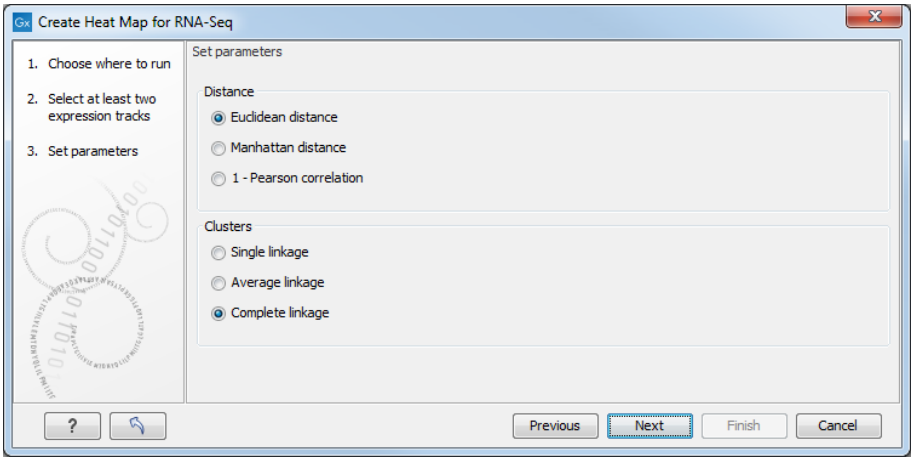


Figure 29.30: *Parameters for Create Heat Map.*

- **1 - Pearson correlation** The Pearson correlation coefficient between two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ is defined as

$$r = \frac{1}{n-1} \sum_{i=1}^{n} (\frac{x_i - \overline{x}}{s_x})(\frac{y_i - \overline{y}}{s_y})$$

  where $\overline{x}/\overline{y}$ is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

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.

After having selected the distance measure, click **Next** to set up the feature filtering options as shown in figure 29.31.

Genomes usually contain too many features to allow for a meaningful visualization of all genes or transcripts. Clustering hundreds of thousands of features is also very time consuming. Therefore it is recommend to reduce the number of features before clustering and visualization.

There are several different **Filter settings** to filter genes or transcripts:

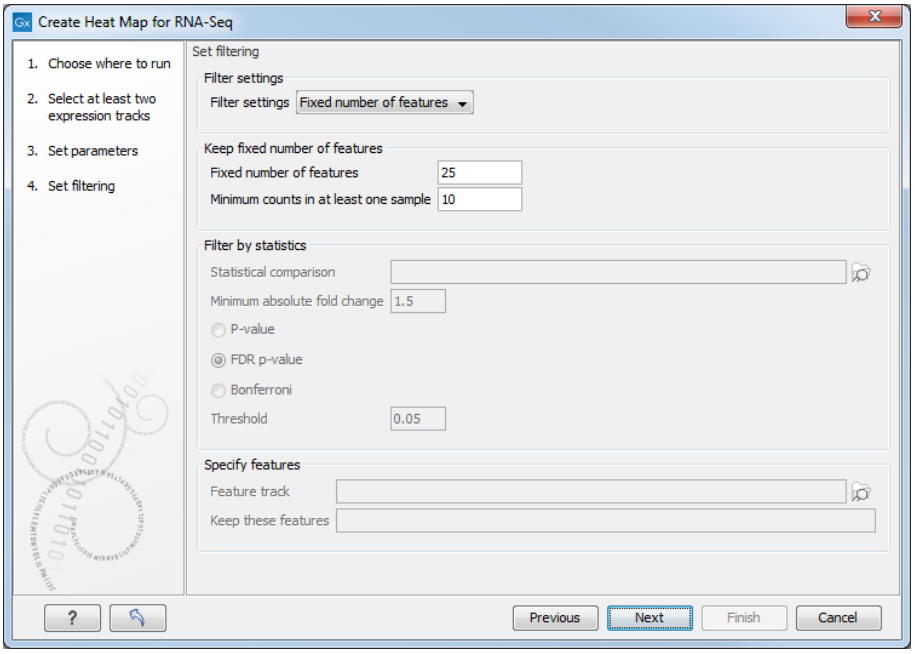


Figure 29.31: *Feature filtering for Create Heat Map.*

- **No filtering** Keeps all features.
- **Fixed number of features**
  - **Fixed number of features** The given number of features with the highest coefficient of variation (the ratio of the standard deviation to the mean) are kept.
  - **Minimum counts in at least one sample** Only features with more than this number of counts in at least one sample will be taken into account. Notice that the counts are raw, un-normalized values.
- **Filter by statistics** Keeps features that are differentially expressed according to the specified cut-offs.
  - **Statistical comparison** A single statistical comparison track output by the Differential Expression for RNA-Seq tool.
  - **Minimum absolute fold change** Only features with a higher absolute fold change are kept.
  - **Threshold** Only features with a lower p-value are kept. It is possible to select which type of p-value to use.
- **Specify features** Keeps a set of features, as specified by either a feature track or by plain text.
  - **Feature track** Any genes or transcripts defined in the feature track will be kept.
  - **Keep these features** A plain text list of feature names. Any white-space characters, and ",", and ";" are accepted as separators.

29.6.2 The heat map view

After the tool completes, a heat map like the one shown in (figure 29.32) is produced. In the heat map each row corresponds to a feature and each column to a sample. The color in the i'th row

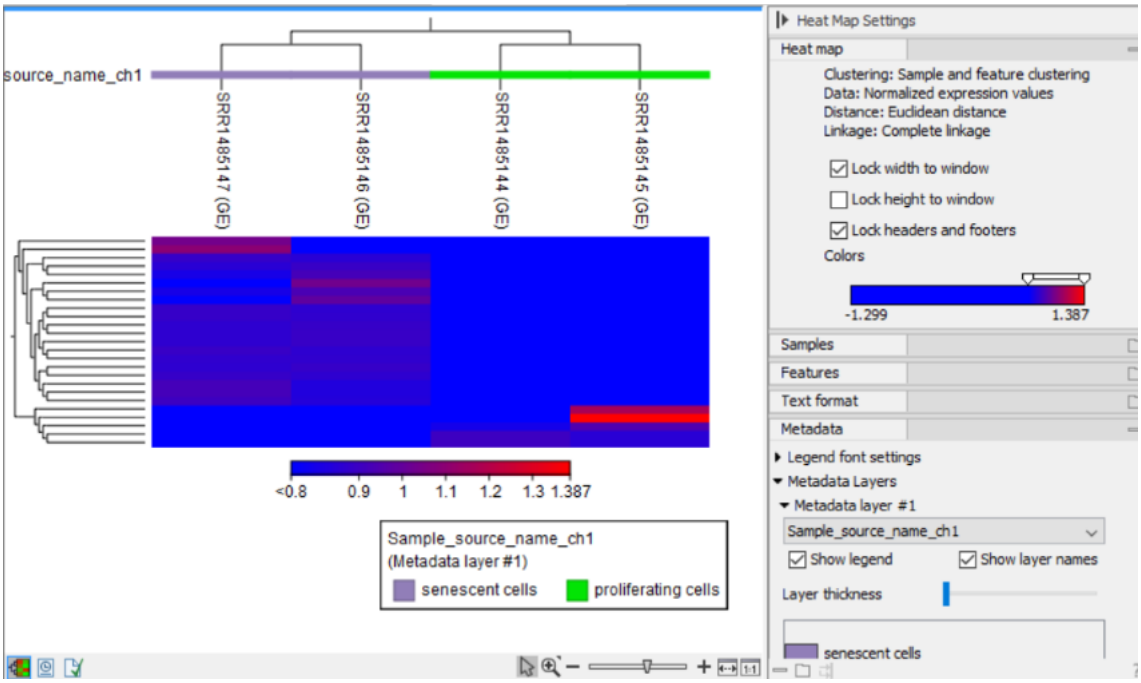


Figure 29.32: *The 2D heat map.*

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 expression values used are normalized log CPM values, see section 29.

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** group (see figure 29.32).

- **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 normally 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 are 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, color legends, and trees above or below the heat map. The tree options also control the **Tree size**, including the option of showing the full tree, no matter how much space it will use.

The **Metadata** group makes it possible to visualize metadata associated with the Expression tracks:

- **Legend font settings** adjusts the label settings.
- **Metadata layers** Adds a color bar to the hierarchical sample tree, colored according to the value of a chosen metadata table column.

29.7 Create Expression Browser

The **Create Expression Browser** tool makes it possible to inspect gene and transcript expression level counts, annotations and statistics for many samples at the same time.

To run the tool:

Toolbox | **RNA-Seq Analysis** | **Create Expression Browser**

Select a number of expression tracks (), either Gene level (GE) or Transcript level Expression (TE) tracks but not a combination of both, and click **Next** (see figure 29.33).

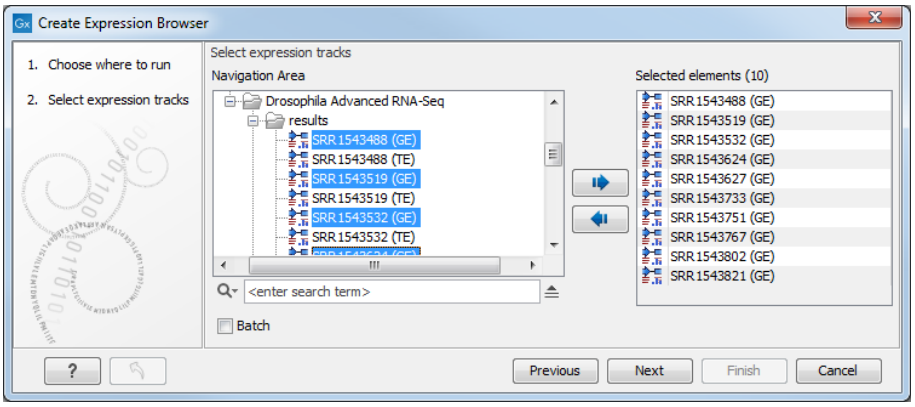


Figure 29.33: *Select expression tracks, either GE or TE.*

In the second wizard dialog (see figure 29.34), you can decide to add one or more statistical comparisons or an annotation source.

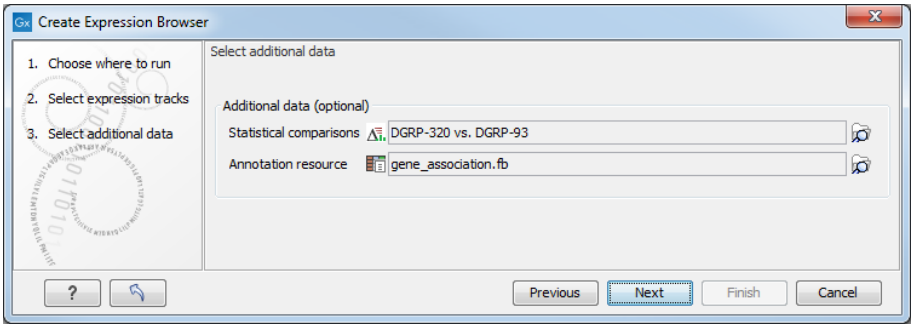


Figure 29.34: *It is optional to provide statistical comparisons and annotation resources.*

Statistical comparisons are generated by the Differential Expression for RNA-Seq tool. You can only choose a comparison that was generated by the same kind of expression tracks as the ones you selected in the previous step. This means that when you want to create an expression browser for GE expression levels, you can only input in the second optional step a statistical comparison generated using GE tracks as well.

Annotation resources can be obtained from different sources:

- In the majority of cases, annotations are obtained from Gene Ontology consortium at http://geneontology.org/page/download-annotations. Download the *.gz file of your choice from the website on to your computer, and import it in the workbench using the Standard Import tool.
- You may have your own annotation data in a spreadsheet. Learn how to import such files here: section J.5.
- Annotations can be generated by the "Blast2GO PRO" plugin (see http://www.qiagenbioinformatics.com/plugins/blast2go-pro/ for more information).
- Some annotations for human, mouse and rat are also bundled together with Reference Data Sets provided in Biomedical Genomics Workbench.

29.7.1 The expression browser

An Expression Browser is shown in figure 29.35.

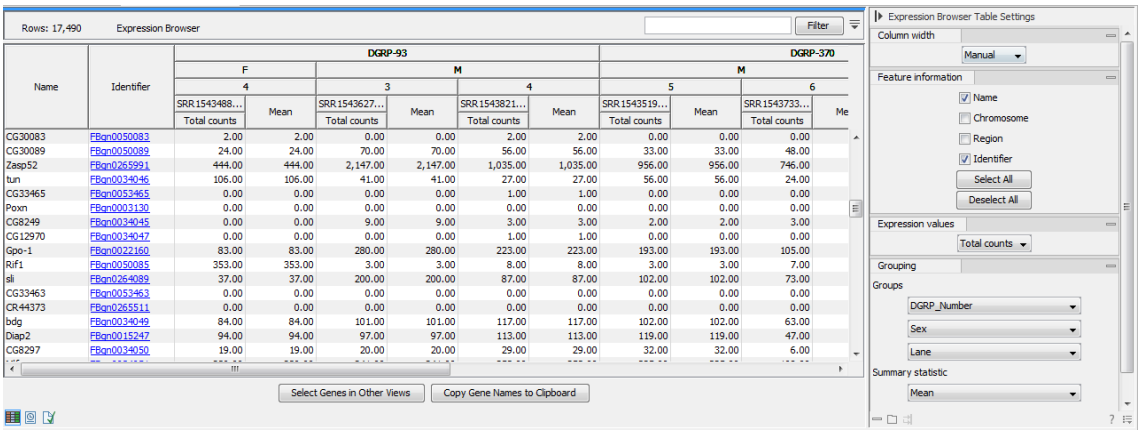


Figure 29.35: *Expression browser table when no statistical comparison or annotations resources were provided.*

Each row represents a gene or a transcript, defined by its name, the chromosome and the region where it is located, as well as an identifier linking to the relevant online database.

The expression values for each sample - or aggregation of samples - can be given by total counts, RPKM, TPM or CPM (Counts Per Million). These measurements differ from each other in two key ways:

1. RPKM and TPM measure the number of *transcripts* whereas total counts and CPM measure the number of *reads*. The distinction is important because in an RNA-Seq experiment, more reads are typically sequenced from longer transcripts than from shorter ones.
2. RPKM, TPM and CPM are normalized for sequencing-depth so their values are *comparable* between samples. Total counts is not normalized, so values are *not comparable* between samples.

How do I get the normalized counts used to calculate fold changes? The CPM expression values are most comparable to the results of the Differential Expression for RNA-Seq tool. However, normalized counts are not used to calculate fold changes; instead the Differential Expression for RNA-Seq tool works by fitting a statistical model (which accounts for differences in sequencing-depth) to raw counts. It is therefore not possible to derive these fold changes from the CPM values by simple algebraic calculations.

It is possible to display the values for individual samples, or for groups of samples as defined by the metadata. Using the drop down menus in the "Grouping" section of the right-hand side setting panel, you can choose to group samples according to up to three metadata layers as shown in figure 29.35.

When individual samples are aggregated, an additional "summary statistic" column can be displayed to give either the mean, the minimum, or the maximum expression value for each group of samples. The table in figure 29.35 shows the mean of the expression values for the first group layer that was selected.

If one or more statistical comparisons are provided, extra columns can be displayed in the table using the "Statistical comparison" section of the Settings panel (figure 29.36). The columns correspond to the different statistical values generated by the "Differential Expression for RNA-seq" as detailed in section 29.5.3.

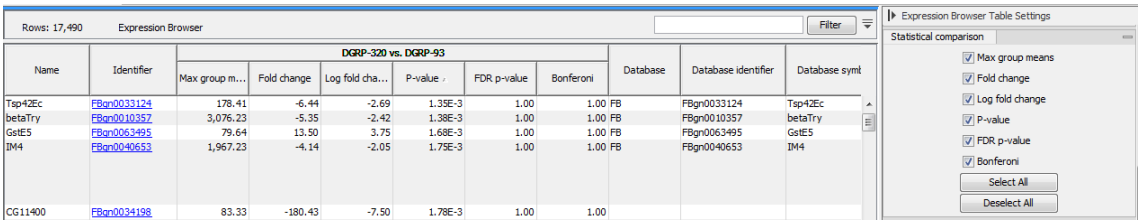


Figure 29.36: *Expression browser table when a statistical comparison is present.*

If an annotation database is provided, extra columns can be displayed in the table using the "Annotation" section of the Settings panel (figure 29.37). Which columns are available depends on the annotation file used. When using a GO annotation file, the GO biological process column will list for each gene or transcript one or several biological processes. Click on the process name to open the corresponding page on the Consortium for Gene Ontology webpage. It is also possible to access additional online information by clicking on the the PMID, RefSeq, HGNC or UniProt accession number when available.

Select the genes of interest and use the button present at the bottom of the table to highlight the genes in other views (volcano plot for instance) or to copy the genes of interest to a clipboard.

29.8 Create Venn Diagram for RNA-Seq

The **Create Venn Diagram** tool makes it possible to compare the overlap of differentially expressed genes or transcripts in two or more statistical comparison tracks. The genes considered to be differentially expressed can be controlled by setting appropriate p-value and fold change thresholds.

To create the Venn diagram:

Toolbox | **RNA-Seq Analysis** | **Create Venn Diagram**

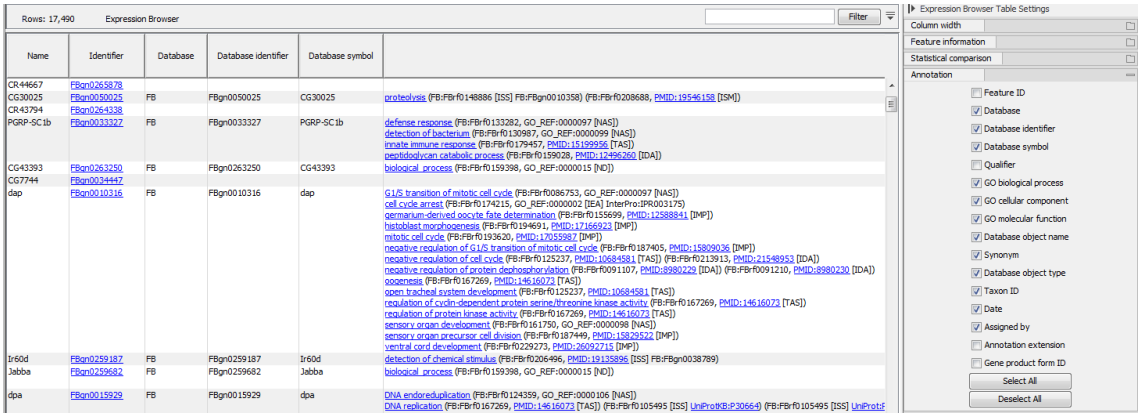


Figure 29.37: *Expression browser table when a GO annotation file is present.*

Select a number of statistical comparison tracks ( ) and click **Next** (see figure 29.38).

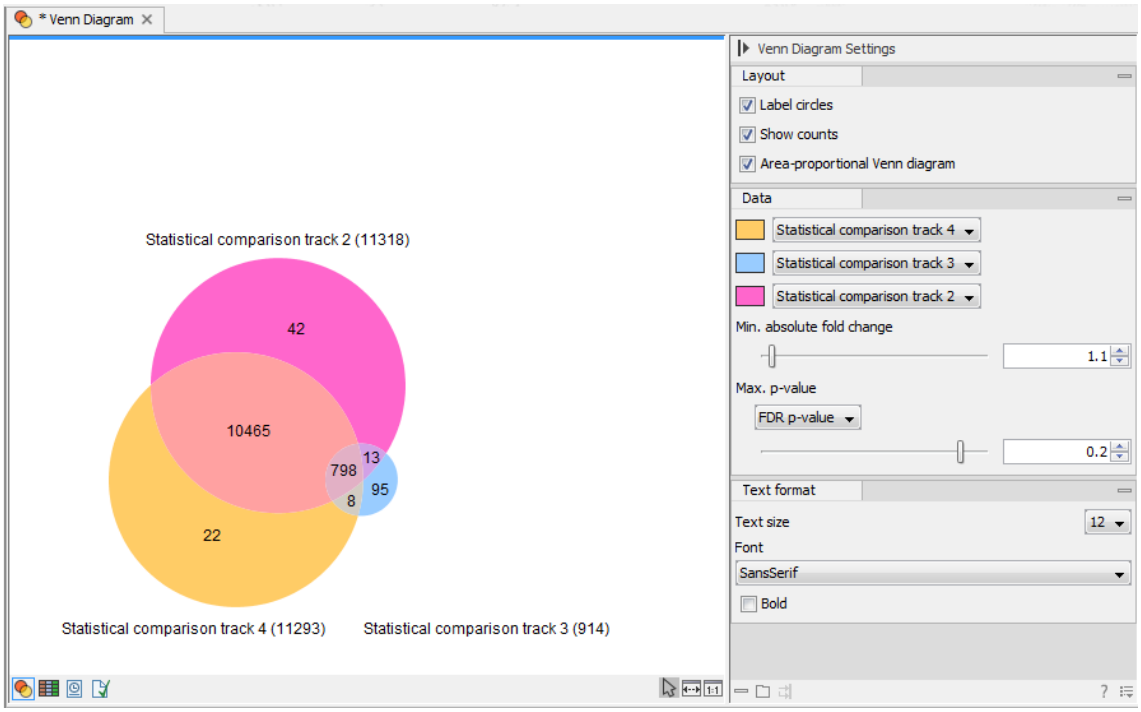


Figure 29.38: *The resulting Venn diagram.*

In the **Side Panel** to the right, it is possible to adjust the Venn Diagram settings. Under **Layout**, you can adjust the general properties of the plot.

- **Label circles** Toggles the names of the statistical comparison tracks.
- **Show counts** Toggles the display of gene or transcript counts.
- **Area-proportional Venn Diagram** When drawn as a Standard Venn Diagram, circles are drawn with fixed positions and identical size. When drawn in the default Venn Diagram mode, sizes and positions of the circles are adjusted in proportion to the number of overlapping features.

The **Data** side panel group makes it possible to choose the differentially expressed genes or

features of interest. The set of statistical comparisons to be compared can be selected using the drop down combo boxes at the top of the group. It is possible to customize the color of a given statistical comparison using the color picker next to the drop down combo box.

- **Minimum absolute fold change** Only genes or transcripts with an absolute fold change higher than the specified threshold are taken into account.
- **Maximum P-value** Only genes or transcripts with a p-value less then the specified threshold will be taken into account. It is possible to select which p-value measure to use.

Finally, the **Text format** group makes it possible to adjust the settings for the count and statistical comparison labels.

29.8.1 Venn diagram table view

It is possible to inspect the p-values and fold changes for each gene or transcript individually in the Venn diagram table view (see figure 29.39).

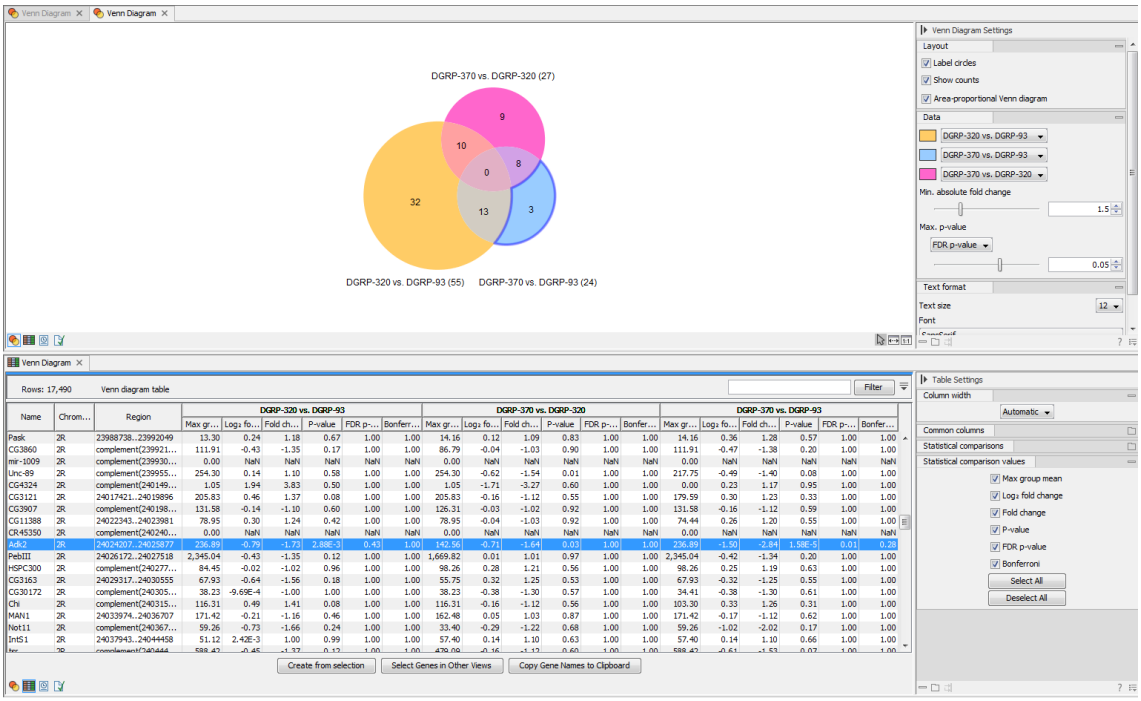


Figure 29.39: *The Venn diagram table view.*

Clicking a circle segment in the Venn Diagram plot will select the genes or transcript in the table view. It is also possible to create a subset list of genes or transcripts using the **Create from selection**.

In the **Side Panel** to the right it is possible to adjust the Table settings. It is possible to adjust the column layout, and select which columns should be included in the table.

29.9 Gene Set Test

The **Gene Set Test** tool tests whether GO terms are over-represented in a set of differentially expressed genes (input as a statistical comparison track) using a hypergeometric test. The tool will require a GO annotation file that must be previously saved in the Navigation Area of the workbench.

GO annotation files are available from several sources (Blast2Go, GO ontology database). Before import, check that the table does have a GO column, and if not, edit the table to change the relevant column header to GO before import in the workbench using the Standard Import function. 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.

RefSeq files are available via the Data Manager, and are saved in the "CLC_Reference" folder in the Navigation Area if you have already downloaded a Reference Data Set.

It is also possible to format a text file of custom annotations into a format the Gene Set Test tool can use (see section J.5.2 and section J.4).

- The file type should be .csv.
- The values should be comma-separated and in quotation marks.
- The first column should be "Probe Set ID" or one of the other recognized values mentioned in the manual, and the values in the first column must match the feature names in the data exactly.
- The actual annotations should be found in one of the "Gene Ontology"-type columns: Gene Ontology Biological Process, Gene Ontology Cellular Component, Gene Ontology Molecular Function.
- The separator // is used to separate the name of an annotation from its description, and the separator /// is used to separate different annotations from each other. Each annotation should then look like: "AnnotationA_name // AnnotationA description /// AnnotationB_name // AnnotationB description".

This custom annotation file can be imported using the Standard Import functionality.

To start the tool:

Toolbox | **RNA-Seq Analysis** | **Gene Set Test**

Select a statistical comparison track () and click **Next** (see figure 29.40). To run several statistical comparisons at once, use the batch function.

In the "Annotation testing parameters" dialog, you need to specify a GO annotation file and have several annotation testing options(see figure 29.41).

- **GOA**: Specify a GO annotation file (such as described in the introduction of this section) using the Browse button to the right of the field.
- **GO biological process** Tests for enriched GO biological processes, i.e., a series of events or molecular functions such as "lipid storage" or "chemical homeostasis".

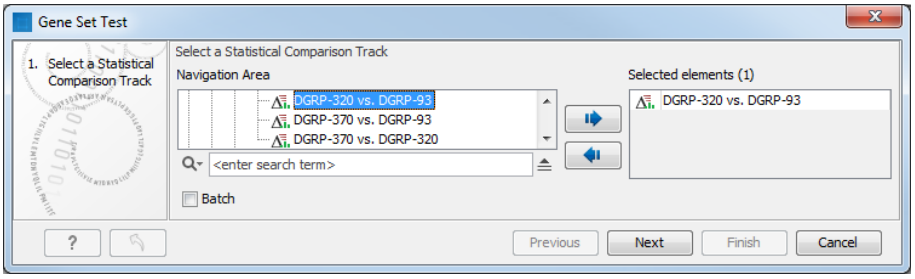


Figure 29.40: *Select one statistical comparison.*

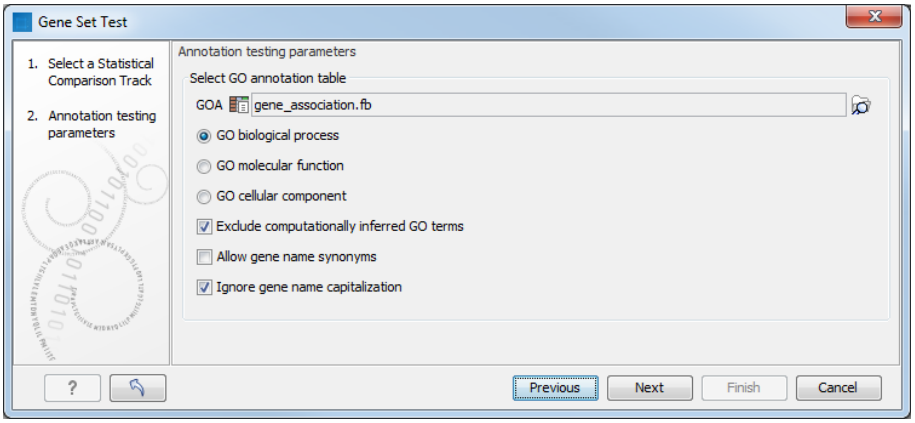


Figure 29.41: *Select annotation tetsing parameters.*

- **GO molecular function** Tests for enriched GO molecular functions. The GO molecular functions of a gene can be such as "retinoic acid receptor activity" or "transcription regulator activity".
- **GO cellular component** Tests for enriched GO cellular component. A GO cellular component ontology describes locations, such as "nuclear inner membrane" or "ubiquitine ligase complex".
- **Exclude computationally inferred GO terms** excludes uncurated GO terms with evidence code IEA, i.e., terms that were automatically curated but not reviewed by a curator.
- **Allow gene name synonyms** allows matching of the gene name with database identifiers and synonyms.
- **Ignore gene name capitalization** ignores capitalization in feature names: a gene called "Dat" in the statistical comparison track will be matched with the one called "dat" in the annotation file when this option is checked. If "Dat" and "dat" are believe to be different genes, the option should be unchecked.

Click Next to access the "Filtering parameters" dialog (see figure 29.42).

Instead of annotating all genes present in the statistical comparison track, it is possible to focus on the subset of genes that are differentially expressed. The filtering parameters allow you to define this subset:

- **Ignore features with mean RPKM below**. Only features where the highest group mean RPKM exceeds this limit will be included in the analysis.

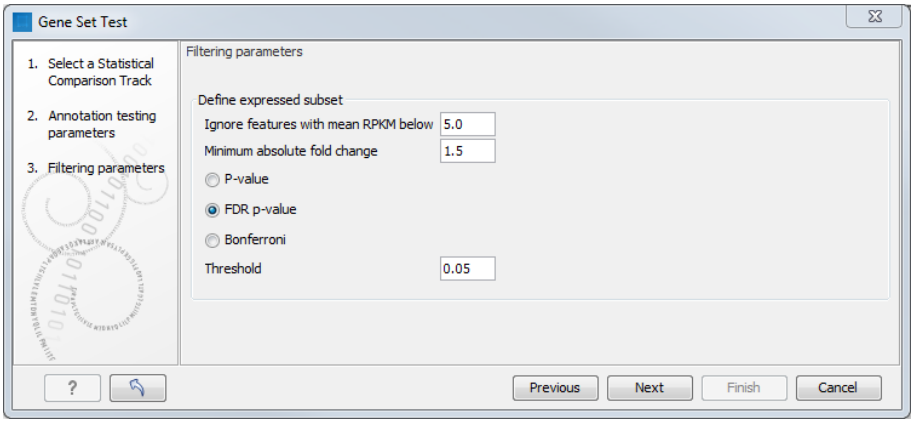


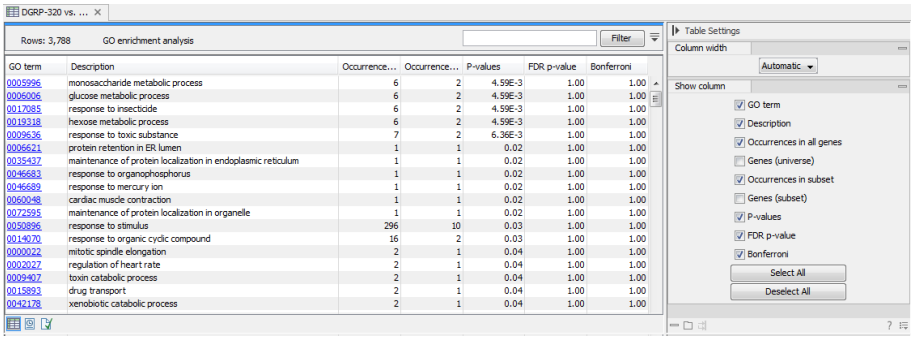
Figure 29.42: *Specify filtering parameters.*

- **Minimum absolute fold change**. Define the minimum absolute fold change value for a feature, and specify whether this fold change should calculated as **p-value**, **FDR p-value** or **Bonferroni** (for a detailed definition of these, see section 30.6.4).
- **Threshold**. Maximum p-value for a feature.

Click **Finish** to Open or Save the file in a specified location of the Navigation Area.

During analysis, a black banner in the left hand side of the workbench warns if duplicate features were found while processing the file. If you get this warning message, consider unchecking the "Ignore gene name capitalization" option.

The output is a table called "GO enrichment analysis" (see figure 29.43). The table is sorted in order of ascending p-values but it can easily be sorted differently, as well as filtered to highlight only the GO terms that are over-represented. The table also provides FDR and Bonferroni-corrected p-values. Note that the p-values provided in the table are meant as a guide, as GO annotations are not strictly independent of each other (for example, "reproduction" is a broad category that encompass a nested set of terms from other categories such as "pheromone biosynthetic process").



GO term	Description	Occurrence...	Occurrence...	P-values	FDR p-value	Bonferroni
0005996	monosaccharide metabolic process	6	2	4.59E-3	1.00	1.00
0006006	glucose metabolic process	6	2	4.59E-3	1.00	1.00
0017085	response to insecticide	6	2	4.59E-3	1.00	1.00
0019318	hexose metabolic process	6	2	4.59E-3	1.00	1.00
0009636	response to toxic substance	7	2	6.36E-3	1.00	1.00
0006621	protein retention in ER lumen	1	1	0.02	1.00	1.00
0035437	maintenance of protein localization in endoplasmic reticulum	1	1	0.02	1.00	1.00
0046683	response to organophosphorus	1	1	0.02	1.00	1.00
0046689	response to mercury ion	1	1	0.02	1.00	1.00
0060048	cardiac muscle contraction	1	1	0.02	1.00	1.00
0072595	maintenance of protein localization in organelle	1	1	0.02	1.00	1.00
0050896	response to stimulus	296	10	0.03	1.00	1.00
0014070	response to organic cyclic compound	16	2	0.03	1.00	1.00
0000022	mitotic spindle elongation	2	1	0.04	1.00	1.00
0002027	regulation of heart rate	2	1	0.04	1.00	1.00
0009407	toxin catabolic process	2	1	0.04	1.00	1.00
0015893	drug transport	2	1	0.04	1.00	1.00
0042178	xenobiotic catabolic process	2	1	0.04	1.00	1.00

Figure 29.43: *The GO enrichment analysis table generated by the Gene Set Test tool.*

Chapter 30

Microarray and Small RNA Analysis tools

Contents

30.1 Small RNA analysis

The small RNA analysis tools in *Biomedical Genomics Workbench* are designed to facilitate trimming of sequencing reads, counting and annotating of the resulting tags using miRBase or other annotation sources and performing expression analysis of the results. The tools are general and flexible enough to accommodate a variety of data sets and applications within small RNA profiling, including the counting and annotation of both microRNAs and other non-coding RNAs from any organism. Illumina, 454 and SOLiD sequencing platforms are supported. For SOLiD, adapter trimming and annotation is done in color space.

The annotation part is designed to make special use of the information in miRBase but more general references can be used as well.

There are generally two approaches to the analysis of microRNAs or other smallRNAs: (1) count the different types of small RNAs in the data and compare them to databases of microRNAs or other smallRNAs, or (2) map the small RNAs to an annotated reference genome and count the numbers of reads mapped to regions which have smallRNAs annotated. The approach taken by *Biomedical Genomics Workbench* is (1). This approach has the advantage that it does not require an annotated genome for mapping — you can use the sequences in miRBase or any other sequence list of smallRNAs of interest to annotate the small RNAs. In addition, small RNAs that would not have mapped to the genome (e.g. when lacking a high-quality reference genome or if the RNAs have not been transcribed from the host genome) can still be measured and their expression be compared. The methods and tools developed for *Biomedical Genomics Workbench* are inspired by the findings and methods described in [Creighton et al., 2009], [Wyman et al., 2009], [Morin et al., 2008] and [Stark et al., 2010].

In the following, the tools for working with small RNAs are described in detail.

30.1.1 Extract and count

First step in the analysis is to import the data (see section 6.3).

The next step is to extract and count the small RNAs to create a *small RNA sample* that can be used for further analysis (either annotating or analyzing using the expression analysis tools):

Toolbox | Microarray and Small RNA Analysis ( ) | Small RNA Analysis ( ) | Extract and Count ( )

This will open a dialog where you select the sequencing reads that you have imported. Click **Next** when the sequencing data is listed in the right-hand side of the dialog. Note that if you have several samples, they should be processed separately.

This dialog (see figure 30.1) is where you specify whether the reads should be trimmed for adapter sequences prior to counting. It is often necessary to trim off remainders of adapter sequences from the reads before counting.

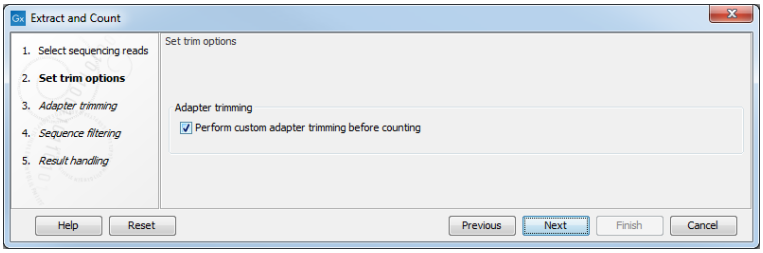


Figure 30.1: *Specifying whether adapter trimming is needed.*

When you click **Next**, you will be able to specify how the trim should be performed as shown in figure 30.2.

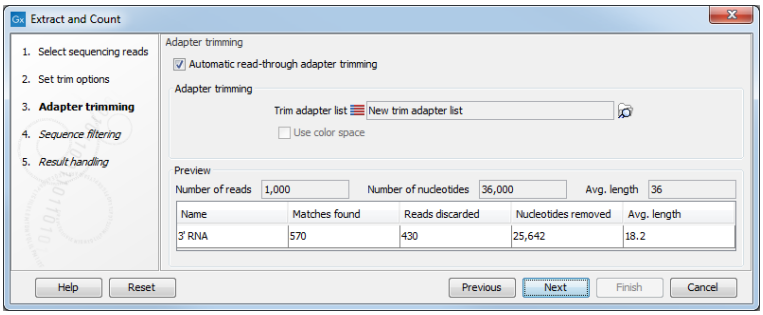


Figure 30.2: *Setting parameters for adapter trim.*

If you have chosen not to trim the reads for adapter sequence, you will see figure 30.3 instead.

The trim options shown in figure 30.2 are the same as described under adapter trim in section 21.2.2. Please refer to this section for more information.

It should be noted that if you expect to see part of adapters in your reads, you would typically choose **Discard when not found** as the action. By doing this, only reads containing the adapter sequence will be counted as small RNAs in the further analysis. If you have a data set where the adapter may be there or not you would choose **Remove adapter**.

Note that all reads will be trimmed for ambiguity symbols such as N before the adapter trim.

Clicking **Next** allows you to specify additional options regarding trimming and counting as shown in figure 30.3.

At the top you can choose to **Trim bases** by specifying a number of bases to be removed from either the 3' or the 5' end of the reads. Below, you can specify the minimum and maximum lengths of the small RNAs to be counted (this is the length after trimming). The minimum length that can be set is 15 and the maximum is 55.

At the bottom, you can specify the **Minimum sampling count**. This is the number of copies of the small RNAs (tags) that are needed in order to include it in the resulting count table (the small

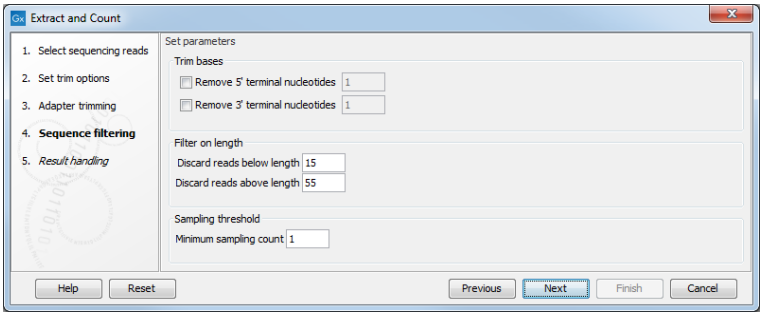


Figure 30.3: *Defining length interval and sampling threshold.*

RNA sample). The actual counting is very simple and relies on **perfect match** between the reads to be counted together[1]. This also means that a count threshold of 1 will include a lot of unique tags as a result of sequencing errors. In order to set the threshold right, the following should be considered:

- If the sample is going to *be annotated*, annotations may be found for the tags resulting from sequencing errors. This means that there is no negative effect of including tags with a low count in the output.
- When using *un-annotated sequences* for discovery of novel small RNAs, it may be useful to apply a higher threshold to eliminate the noise from sequencing errors. However, this can be done at a later stage by filtering the sample and creating a sub-set.
- When multiple samples are compared, it is interesting to know if one tag which is abundant in one sample is also found in another, even at a very low number. In this case, it is useful to include the tags with very low counts, since they may become more trustworthy in combination with information from other samples.
- Setting the count threshold higher will reduce the size of the sample produced which will reduce the memory and disk usage when working with the results.

Clicking **Next** allows you to specify the output of the analysis as shown in 30.4.

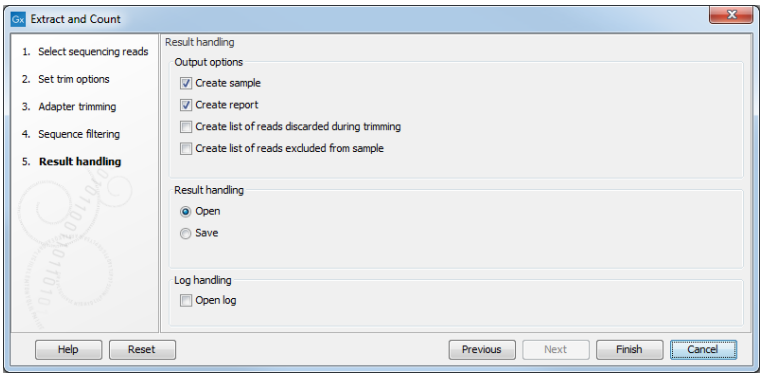


Figure 30.4: *Output options.*

The options are:

[1]Note that you can identify variants of the same miRNA when annotating the sample (see below).

Create sample This is the primary result showing all the tags and respective counts (an example is shown in figure 30.5). Each row represents a tag with the actual sequence as the feature ID and columns with **Expression values**, **Length**, and **Count**. Expression values and the actual count are based on 100 % similarity[2]. The sample can be used in further analysis in the "raw" form, or you can annotate it (see below). The tools for working with the data in the sample are described in section 30.1.4. Note that when small RNA samples are used for setting up and experiment, it is always the Expression values that will be used.

Create report This will create a summary report as described below.

Create list of reads discarded during trimming This list contains the reads where no adapter was found (when choosing **Discard when not found** as the action).

Create list of reads excluded from sample This list contains the reads that passed the trimming but failed to meet the sampling thresholds regarding minimum/maximum length and number of copies.

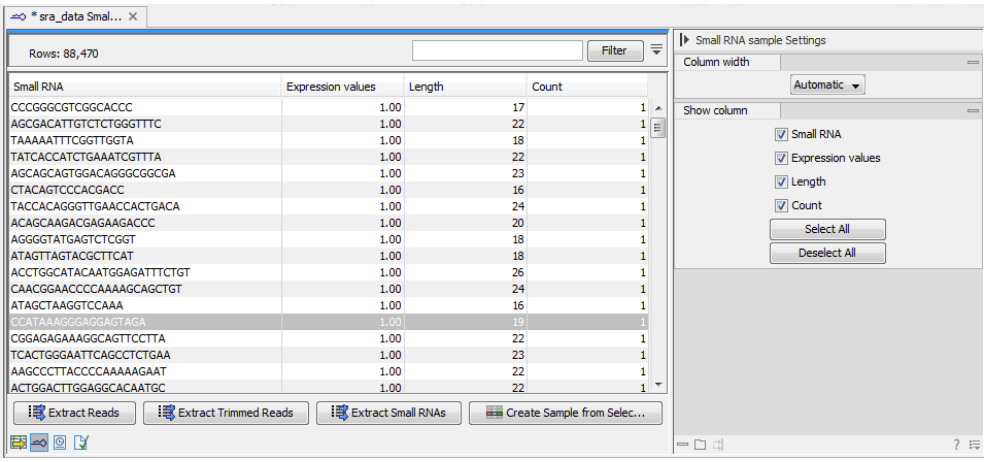


Figure 30.5: *The tags have been extracted and counted.*

The summary report includes the following information (an example is shown in figure 30.6):

Trim summary Shows the following information for each input file:

- Number of reads in the input.
- Average length of the reads in the input.
- Number of reads after trim. The difference between the number of reads in the input and this number will be the number of reads that are discarded by the trim.
- Percentage of the reads that pass the trim.
- Average length after trim. When analyzing miRNAs, you would expect this number to be around 22. If the number is significantly lower or higher, it could indicate that the trim settings are not right. In this case, check that the trim sequence is correct, that the strand is right, and adjust the alignment scores. Sometimes it is preferable to increase the minimum scores to get rid of low-quality reads. The average length after trim could also be somewhat larger than 22 if your sequenced data contains a mixture of miRNA and other (longer) small RNAs.

[2]Note that you can identify variants of the same miRNA when annotating the sample (see below).

Read length before/after trimming Shows the distribution of read lengths before and after trim. The graph shown in figure 30.6 is typical for miRNA sequencing where the read lengths after trim peaks at 22 bp.

Trim settings The trim settings summarized. Note that ambiguity characters will automatically be trimmed.

Detailed trim results This is described in the Trim output section 21.2.5.

Tag counts The number of tags and two plots showing on the x-axis the counts of tags and on the y-axis the number of tags for which this particular count is observed. The plot is in a zoomed version where only the lower part of the y-axis is shown to make it possible to see the numbers of tags higher counts.

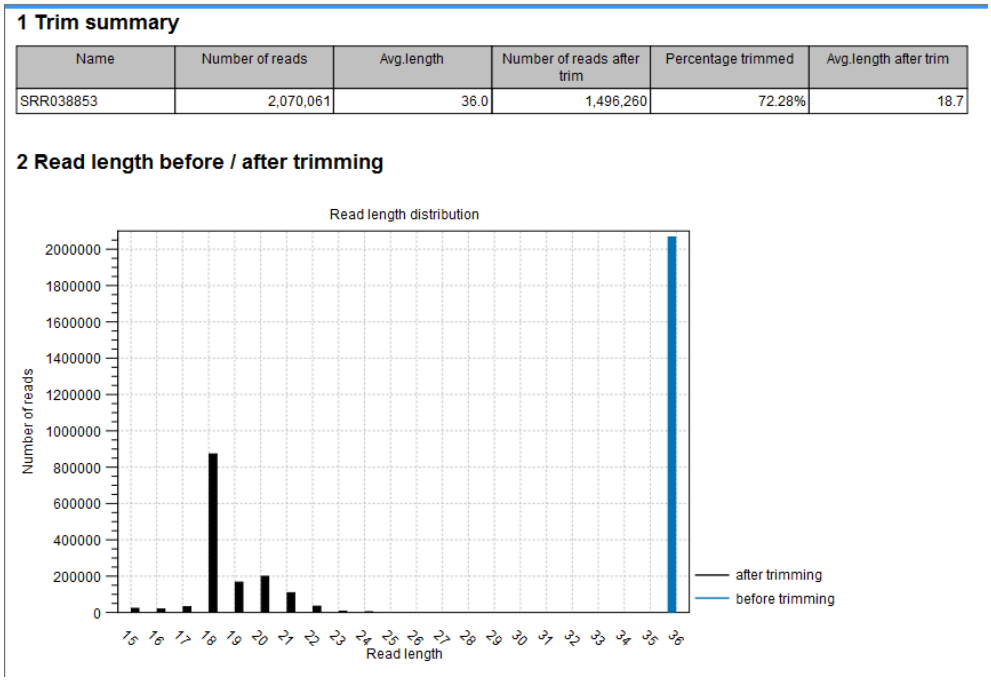


Figure 30.6: *A summary report of the counting.*

30.1.2 Downloading miRBase

In order to make use of the additional information about mature regions on the precursor miRNAs in miRBase, you need to use the integrated tool to download miRBase rather than downloading it from `http://www.mirbase.org/`:

> **Toolbox** | **Microarray and Small RNA** | **Small RNA Analysis ( )** | **Download miRBase ( )**

This will download a sequence list with all the precursor miRNAs including annotations for mature regions. The list can then be selected when annotating the samples with miRBase (see section 30.1.3).

The downloaded version will always be the latest version (it is downloaded from `ftp://mirbase.org/pub/mirbase/CURRENT/miRNA.dat.gz`). Information on the version number of miRBase is also available in the **History** ( ) of the downloaded sequence list, and when using this for annotation, the annotated samples will also include this information in their **History** ( ).

Importing the miRBase data file You can also import the miRBase data file directly into the Workbench. The file can be downloaded from ftp://mirbase.org/pub/mirbase/CURRENT/miRNA.dat.gz.

In order for the file to be recognized as a miRBase file, you have to select `miRBase dat` in the **Force import as type** menu of the Standard Import dialog.

A *.dat file follows the following format:

```
ID cel-let-7
XX
DE Caenorhabditis elegans let-7 stem-loop
XX
FH Key Location/Qualifiers
FH
FT miRNA 17..38
FT /product="cel-let-7-5p"
FT miRNA 60..81
FT /product="cel-let-7-3p"
XX
SQ Sequence 99 BP; 26 A; 19 C; 24 G; 0 T; 30 other;
uacacugugg auccggugag guaguagguu guauaguuug gaauauuacc accggugaac 60
uaugcaauuu ucuaccuuac cggagacaga acucuucga 99
//
ID cel-lin-4
XX
DE Caenorhabditis elegans lin-4 stem-loop
XX
FH Key Location/Qualifiers
FH
FT miRNA 16..36
FT /product="cel-lin-4-5p"
FT miRNA 55..76
FT /product="cel-lin-4-3p"
XX
SQ Sequence 94 BP; 17 A; 25 C; 26 G; 0 T; 26 other;
augcuuccgg ccuguucccu gagaccucaa gugugagugu acuauugaug cuucacaccu 60
gggcucuccg gguaccagga cgguuugagc agau 94
//
```

Creating your own miRBase file If you wish to construct a file yourself to be used as a miRBase file for annotation, this is also possible if you format the file in the same way as the miRBase data file. In particular, the following needs to be in place:

- The sequences needs "miRNA" annotation on the precursor sequences. In the Workbench, you can add a miRNA annotation by selecting a region and right click to **Add Annotation**. You should have a max 2 miRNA annotations per precursor sequences. Matches to first

miRNA annotation are counting in 5′ column. Matches to second miRNA annotation are counted as 3′ matches.

- If you have sequence list containing sequences from multiple species, the **Latin name** of the sequences should be set. This is used in the annotation dialog (see section 30.1.3) where you can select the species. If the Latin name is not set, the dialog will show "N/A".

Once you have created the file, it has to be imported as described above.

30.1.3 Annotating and merging small RNA samples

The small RNA sample produced when counting the tags (see section 30.1.1) can be enriched by *Biomedical Genomics Workbench* by comparing the tag sequences with annotation resources such as miRBase and other small RNA annotation sources. Note that the annotation can also be performed on an experiment, set up from small RNA samples (see section 30.2.1).

Besides adding annotations to known small RNAs in the sample, it is also possible to merge variants of the same small RNA to get a cumulative count. When initially counting the tags, the Workbench requires that the trimmed reads are identical for them to be counted as the same tag. However, you will often see different variants of the same miRNA in a sample, and it is useful to be able to count these together. This is also possible using the tool to annotate and merge samples.

Toolbox | Microarray and Small RNA Analysis | Small RNA Analysis ( ) | Annotate and Merge Counts ( )

This will open a dialog where you select the small RNA samples ( ) to be annotated. Note that if you have included several samples, they will be processed separately but summarized in one report providing a good overview of all samples. You can also input **Experiments** ( ) (see section 30.2.1) created from small RNA samples. Click **Next** when the data is listed in the right-hand side of the dialog.

This dialog (figure 30.7) is where you define the annotation resources to be used.

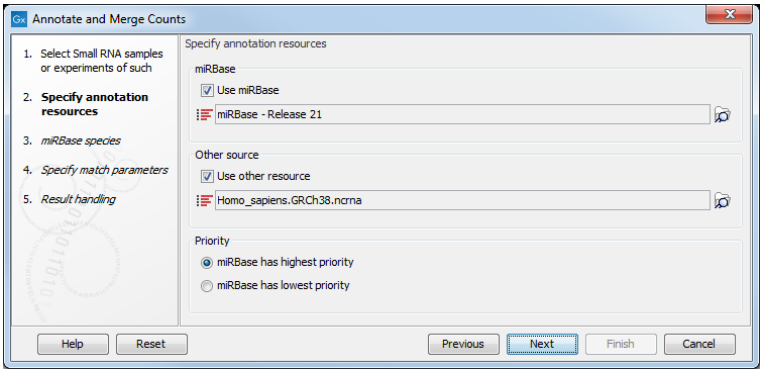


Figure 30.7: *Defining annotation resources.*

There are two ways of providing annotation sources:

- Downloading miRBase using the integrated download tool (explained in section 30.1.2).
- Importing a list of sequences, e.g. from a fasta file. This could be from Ensembl, e.g. ftp://ftp.ensembl.org/pub/release-57/fasta/homo_sapiens/ncrna/

Homo_sapiens.GRCh37.57.ncrna.fa.gz or from ncRNA.org: http://togodb.biosciencedbc.jp/togodb/view/frnadb_summary.

Note: We recommend using the integrated download tool to import miRBase. Although it is possible to import it as a fasta file, the same options with regards to species will not be available if you import from a file.

The downloaded miRBase file contains all precursor sequences from the latest version of miRBase http://www.mirbase.org/ including annotations defining the mature regions (see an example in figure 30.8).

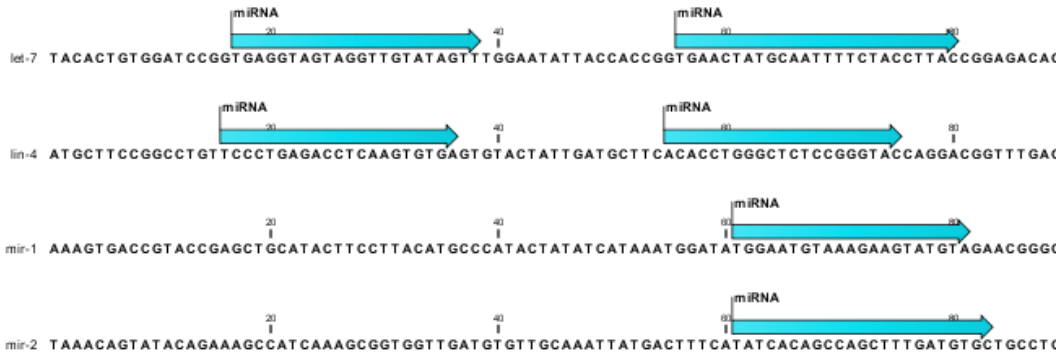


Figure 30.8: *Some of the precursor miRNAs from miRBase have both 3' and 5' mature regions (previously referred to as mature and mature*) annotated (as the two first in this list).*

This means that it is possible to have a more fine-grained classification of the tags using miRBase compared to a simple fasta file resource containing the full precursor sequence. This is the reason why the miRBase annotation source is specified separately in figure 30.7.

At the bottom of the dialog, you can specify whether miRBase should be prioritized over the additional annotation resource. The prioritization is explained in detail later in this section. To prioritize one over the other can be useful when there is redundant information (e.g. if you have an additional source that also contains all the miRNAs from miRBase and you prefer the miRBase annotations when possible).

When you click **Next**, you will be able to choose which species from miRBase should be used and in which order (see figure 30.9). Note that if you have not selected a miRBase annotation source, you will go directly to the next step shown in figure 30.10.

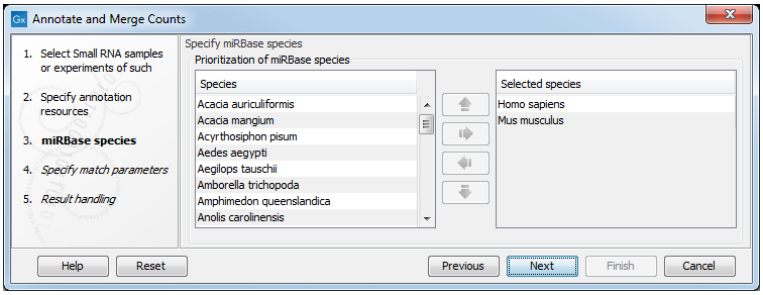


Figure 30.9: *Defining and prioritizing species in miRBase.*

To the left, you see the list of species in miRBase. This list is dynamically created based on the information in the miRBase file. Using the arrow button ( ) you can add species to the right-hand

panel. The order of the species is important since the tags are annotated iteratively based on the order specified here. This means that in the example in figure 30.9, a human miRNA will be preferred over mouse, even if they are identical in sequence (the prioritization is elaborated below). The up and down arrows ( )/ ( ) can be used to change the order of species.

When you click **Next**, you will be able to specify how the alignment of the tags against the annotation sources should be performed (see figure 30.10).

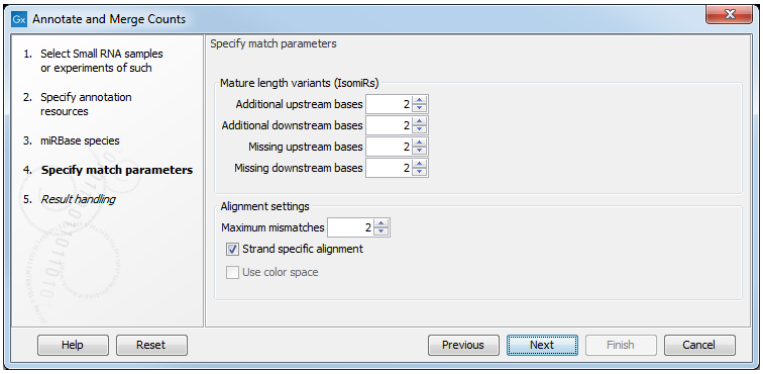


Figure 30.10: *Setting parameters for aligning.*

The panel at the top is active only if you have chosen to annotate with miRBase. It is used to define the requirements to the alignment of a read for it to be counted as a 3' or 5' mature region (previously referred to as mature and mature*) tag:

Additional upstream bases This defines how many bases the tag is allowed to extend the annotated mature region at the 5' end and still be categorized as mature.

Additional downstream bases This defines how many bases the tag is allowed to extend the annotated mature region at the 3' end and still be categorized as mature.

Missing upstream bases This defines how many bases the tag is allowed to miss at the 5' end compared to the annotated mature region and still be categorized as mature.

Missing downstream bases This defines how many bases the tag is allowed to miss at the 3' end compared to the annotated mature region and still be categorized as mature.

At the bottom of the dialog you can specify the **Maximum mismatches** (default value is 2). Furthermore, you can specify if the alignment and annotation should be performed in **color space** which is available when your small RNA sample is based on SOLiD data. [3] Finally, you can choose whether the tags should be aligned against both strands of the reference or only the positive strand. Usually it is only necessary to align against the positive strand.

At this point, a more elaborate explanation of the annotation algorithm is needed. The short read mapping algorithm in the *Biomedical Genomics Workbench* is used to map all the tags to the reference sequences which comprise the full precursor sequences from miRBase and the sequence lists chosen as additional resources. The mapping is done in several rounds: the first round is done requiring a perfect match, the second allowing one mismatch, the third

[3] Note that this option is only going to make a difference for tags with low counts. Since the actual tag counting in the first place is done based on perfect matches, the highly abundant tags are not likely to have sequencing errors, and aligning in color space does not add extra benefit for these.

allowing two mismatches etc. No gaps are allowed. The number of rounds depend on the number of mismatches allowed[4] (default is two which means three rounds of read mapping, see figure 30.10).

After each round of mapping, the tags that are mapped will be removed from the list of tags that continue to the next round. This means that a tag mapping with perfect match in the first round will not be considered for the subsequent one-mismatch round of mapping.

Note: If there are mismatches in the read, there will be a limitation on how short reads can be mapped:

- A minimum read length of 17 nucleotides is required when the read contains one mismatch.
- A minimum read length of 21 nucleotides is required when the read contains two mismatches.
- A minimum read length of 25 nucleotides is required when the read contains three mismatches.

Following the mapping, the tags are classified into the following categories according to where they match.

- Mature 5' exact
- Mature 5' super
- Mature 5' sub
- Mature 5' sub/super
- Mature 3' exact
- Mature 3' super
- Mature 3' sub
- Mature 3' sub/super
- Precursor
- Other

All these categories except *Other* refer to hits in miRBase. For hits on mirBase sequences we distinguish between where on the sequences the tags match. The mirBase sequences may have up to two mature micro RNAs annotated. We refer to a mature miRNA that is located closer (or equally close) to the 5' end than to the 3' end as 'Mature 5''. A mature miRNA that is located closer to the 3' end is referred to as 'Mature 3''. *Exact* means that the tag matches exactly to the annotated mature 5' or 3' region; *Sub* means that the observed tag is shorter than the annotated mature 5' or mature 3'; *super* means that the observed tag is longer than the annotated mature 5' or mature 3'. The combination *sub/super* means that the observed tag extends the annotation in one end and is shorter at the other end. Precursor means that the tag matches on a mirBase

[4]For color space, the maximum number of mismatches is 2.

sequence, but not within the extended annotated mature region(s). These are defined by the "mature length variants (IsomiRs)" parameters in the "specify match parameters" wizard step (by default these parameters are set to 2 which means that reads that are at most 2 bases too long or too short relative to the annotated mature region are all considered mature hits). The Other category is for hits in the other resources (the information about resource is also shown in the output).

An example of an alignment is shown in figure 30.11 using the same alignment settings as in figure 30.10.

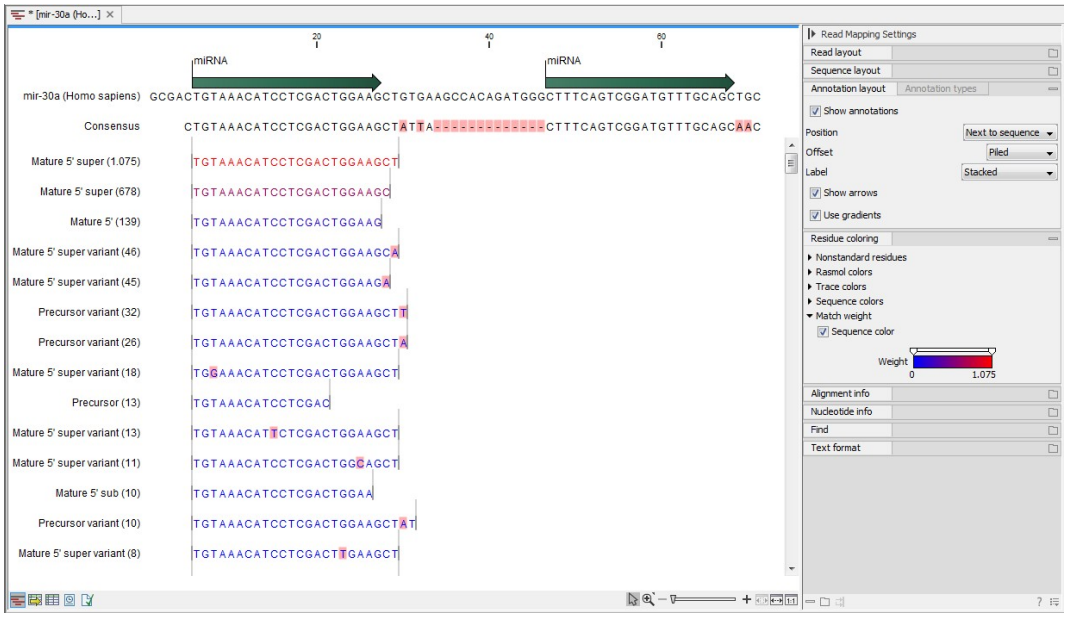


Figure 30.11: *Alignment of length variants of mir-30a.*

The two tags at the top are both classified as *mature 5' super* because they cover and extend beyond the annotated mature 5' RNA. The third tag is identical to the annotated mature 5'.

If a tag has several hits, the list above is used for prioritization. This means that for example, a *Mature 5' sub* is preferred over a *Mature 3' exact*. Note that if miRBase was chosen as lowest priority (figure 30.7), the *Other* category will be at the top of the list. All tags mapping to a miRBase reference without qualifying to any of the mature 5' and mature 3' types will be typed as *Other*. Also note that if a tag has several hits to references *with the same priority* (e.g. the tag matches the mature regions of two different miRBase sequences) it will be annotated with all these sequences. In the report we refer to these tags as 'ambiguously annotated'.

In case you have selected more than one species for miRBase annotation (e.g. Homo Sapiens and Mus Musculus) the following rules for adding annotations apply:

1. If a tag has hits with the same priority for both species, the annotation for the top-prioritized species will be added.

2. Read category priority is stronger than species category priority: If a read is a higher priority match for a mouse miRBase sequence than it is for a human miRBase sequence the annotation for the mouse will be used

Clicking **Next** allows you to specify the output of the analysis as shown in 30.12.

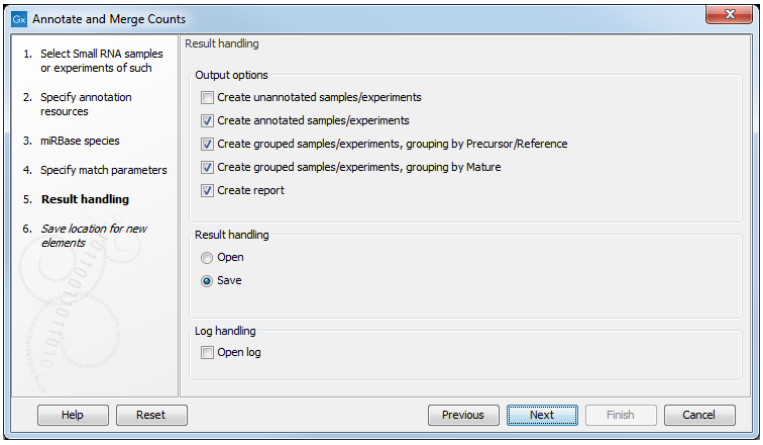


Figure 30.12: *Output options.*

The options are:

Create unannotated sample All the tags where no hit was found in the annotation source are included in the unannotated sample. This sample can be used for investigating novel miRNAs, see section 30.1.5. No extra information is added, so this is just a subset of the input sample.

Create annotated sample This will create a sample as described in section 30.1.4. In this sample, the following columns have been added to the counts.

Name This is the name of the annotation sequence in the annotation source. For miRBase, it will be the names of the miRNAs (e.g. *let-7g* or *mir-147*), and for other source, it will be the name of the sequence.

Resource This is the source of the annotation, either miRBase (in which case the species name will be shown) or other sources (e.g. Homo_sapiens.GRCh37.57.ncrna).

Match type The match type can be exact or variant (with mismatches) of the following types:

- Mature 5'
- Mature 5' super
- Mature 5' sub
- Mature 5' sub/super
- Mature 3'
- Mature 3' super
- Mature 3' sub
- Mature 3' sub/super
- Other

Mismatches The number of mismatches.

Note that if a tag has two equally prioritized hits, they will be shown with // between the names. This could be e.g. two precursor sequences sharing the same mature sequence (also see the sample grouped on mature below).

Create grouped sample, grouping by Precursor/Reference This will create a sample as described in section 30.1.4. All variants of the same reference sequence will be merged to create one expression value for all.

Expression values. The expression value can be changed at the bottom of the table. The default is to use the counts in the mature 5' column.

Name. The name of the reference. For miRBase this will then be the name of the precursor.

Resource. The name of the resource that the reference comes from.

Exact mature 5'. The number of exact mature 5' reads.

Mature 5'. The number of all mature 5' reads including sub, super and variants.

Unique exact mature 5'. In cases where one tag has several hits (as denoted by the // in the ungrouped annotated sample as described above), the counts are distributed evenly across the references. The difference between *Exact mature 5'* and *Unique exact mature 5'* is that the latter only includes reads that are unique to this reference.

Unique mature 5'. Same as above but for all mature 5's, including sub, super and variants.

Exact mature 3'. Same as above, but for mature 3'.

Mature 3'. Same as above, but for mature 3'.

Unique exact mature '3. Same as above, but for mature 3'.

Unique mature '3. Same as above, but for mature 3'.

Exact other. Exact matches in miRBase sequences, but outside annotated mature regions.

Other. All matches in miRBase sequences, but outside annotated mature regions, including variants.

Total. The total number of tags mapped and classified to the precursor/reference sequence.

Note that, for non-mirBase sequences, the counts are collected in the 'Mature 5'' columns: 'Exact mature 5'' (number reads that map to the sequence without mismatches), 'Mature 5' (number reads that map to the sequence, including those with mismatches), 'Unique exact mature 5' (number reads that map uniquely to the sequence without mismatches) and 'Unique mature 5' (number reads that map uniquely to the sequence, including those with mismatches).

Create grouped sample, grouping by Mature This will create a sample as described in section 30.1.4. This is also a grouped sample, but in addition to grouping based on the same reference sequence, the tags in this sample are grouped on the same mature 5'. This means that two precursor variants of the same mature 5' miRNA are merged. For precursor miRNAs that have both a 5' and a 3', we group on both 5' mature and 3' mature. For these miRNAs you will see two rows in the grouped on mature output table, one with the 5' mature sequence as feature ID and one with the 3' mature sequence as feature ID. The 'Match type' column indicates whether the feature ID is 5' or 3' mature. Note that it is only possible to create this sample when using miRBase as annotation resource (because the Workbench has a special interpretation of the miRBase annotations for mature as described previously). To find identical mature 5' miRNAs, the Workbench compares all the mature 5' sequences and when they are identical, they are merged. The names of the precursor sequences merged are all shown in the table.

Expression values. The expression value can be changed at the bottom of the table. The default is to use the counts in the mature 5' column.

Name. The name of the reference. When several precursor sequences have been merged, all the names will be shown separated by //.

Resource. The species of the reference.

Exact mature 5'. The number of exact mature 5' reads.

Mature 5'. The number of all mature 5' reads including sub, super and variants.

Unique exact mature 5'. In cases where one tag has several hits (as denoted by the // in the ungrouped annotated sample as described above), the counts are distributed evenly across the references. The difference between *Exact mature 5'* and *Unique exact mature 5'* is that the latter only includes reads that are unique to one of the precursor sequences that are represented under this mature 5' sequence.

Unique mature 5'. Same as above but for all mature 5's, including sub, super and variants.

Create report. A summary report described below.

The summary report includes the following information (an example is shown in figure 30.13):

1 Summary

Name	Small RNAs	Annotated	Percentage	Ambiguously annotated	Percentage	Reads	Annotated	Percentage	Ambiguously annotated	Percentage
sra_data Small RNA sample	88.470	32.125	36,3%	9.788	11,1%	1.720.280	1.510.721	87,8%	971.902	56,5%

2 Resources

Resource	Sequences in resource	Sequences found	Percentage found
miRBase (Homo sapiens)	1.600	599	37,4%
miRBase (Mus musculus)	855	84	9,8%
Homo_sapiens.GRCh37.57.ncrna	12.887	3.655	28,4%

3 Reads

Annotation	Count	Percentage	Perfect matches	%	1 mismatch	%	2 mismatches	%
Annotated	1.510.721	87,8%	1.212.258	80,2%	247.484	16,4%	50.979	3,4%
- with miRBase	1.467.902	97,2%	1.188.128	80,9%	234.583	16,0%	45.191	3,1%
- Homo sapiens	1.456.045	99,2%	1.182.700	81,2%	230.445	15,8%	42.900	2,9%
- Mus musculus	11.857	0,8%	5.428	45,8%	4.138	34,9%	2.291	19,3%
- with Homo_sapiens. GRCh37.57. ncrna	42.819	2,8%	24.130	56,4%	12.901	30,1%	5.788	13,5%
Unannotated	209.559	12,2%						
Total	1.720.280	100,0%						

4 Small RNAs

Annotation	Count	Percentage
Annotated	32.125	36,3%
- with miRBase	21.490	66,9%
- Homo sapiens	20.259	94,3%
- Mus musculus	1.231	5,7%
- with Homo_sapiens.GRCh37.57.ncrna	10.635	33,1%
Unannotated	56.345	63,7%
Total	88.470	100,0%

Figure 30.13: *A summary report of the annotation.*

Summary Shows the following information for each input sample:

- Number of small RNAs(tags) in the input.

- Number of annotated tags (number and percentage).
- Number of ambiguously annotated tags (number and percentage).
- Number of reads in the sample (one tag can represent several reads)
- Number of annotated reads (number and percentage).
- Number of ambiguously annotated reads (number and percentage).

Resources Shows how many matches were found in each resource:

- Number of sequences in the resource.
- Number of sequences where a match was found (i.e. this sequence has been observed at least once in the sequencing data).

Reads Shows the number of reads that fall into different categories (there is one table per input sample). On the left hand side are the annotation resources. For each resource, the count and percentage of reads in that category are shown. Note that the percentage are relative to the overall categories (e.g. the miRBase reads are a percentage of all the *annotated* reads, not all reads). This is information is shown for each mismatch level.

Small RNAs Similar numbers as for the reads but this time for each small RNA tag and without mismatch differentiation.

Read count proportions A histogram showing, for each interval of read counts, the proportion of annotated (respectively, unannotated) small RNAs with a read count in that interval. Annotated small RNAs may be expected to be associated with higher counts, since the most abundant small RNAs are likely to be known already.

Annotations (miRBase) Shows an overview table for classifications of the number of reads that fall in the miRBase categories for each species selected.

Annotations (Other) Shows an overview table with read numbers for total, exact match and mutant variants for each of the other annotation resources.

An example of the results table is shown at the bottom of figure 30.14.

In the upper part of the figure, you can see two tables showing the imported miRBase file (opened twice for illustration purpose). This example has been included to illustrate the logic behind the order in which the resources are listed. If you look at the results table found at the bottom of the figure, you will see that in the **Resource** column, Mus musculus is listed before Homo sapiens in the first row and in the second row Homo sapiens is listed before Mus musculus. The rationale behind this can be seen from the two other tables if you look at the accession numbers. If you filter only to include Homo sapiens and Mus musculus RNAs, and sort on the "name" column, you can see that for mir-130b (Mus musculus/Homo sapiens), the miRBase number of the mouse RNA is smaller than the miRBase number of the human RNA. Conversely, for mir-495 (Homo sapiens/Mus musculus), the number is smaller in the human case.

30.1.4 Working with the small RNA sample

Generally speaking, the small RNA sample comes in two variants:

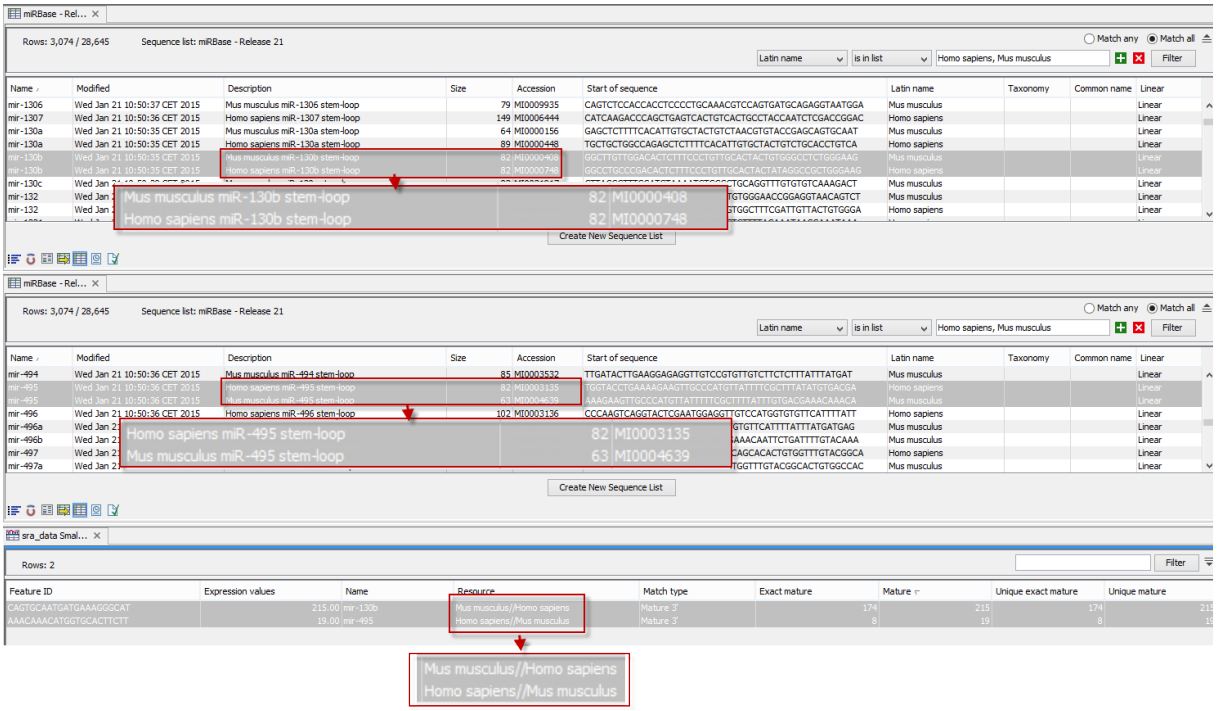


Figure 30.14: *The results table (at the bottom of the figure) shown in split view with the imported miRBase data shown twice to illustrate that the order in which the resource is listed is determined by the miRBase accession number. Due to the very small fonts in the table, the most important information has been magnified (red box).*

- The *un-grouped* sample, either as it comes directly from the **Extract and Count** ( ) or when it has been annotated. In this sample, there is one row per tag, and the feature ID is the tag sequence.
- The *grouped* sample created using the **Annotate and Merge Counts** ( ) tool. In this sample, each row represents several tags grouped by a common Mature or Precursor miRNA or other reference.

Below, these two kinds of samples are described in further detail. Note that for both samples, filtering and sorting can be applied, see section 3.3.

The un-grouped sample

An example of an un-grouped annotated sample is shown in figure 30.15.

By selecting one or more rows in the table, the buttons at the bottom of the view can be used to extract sequences from the table:

Extract Reads ( ) This will extract the original sequencing reads that contributed to this tag. Figure 30.16 shows an example of such a read. The reads include trim annotations (for use when inspecting and double-checking the results of trimming). Note that if these reads are used for read mapping, the trimmed part of the read will automatically be removed. If all rows in the sample are selected and extracted, the sequence list would be the same as the input except for the reads that did not meet the adapter trim settings and the sampling thresholds (tag length and number of copies).

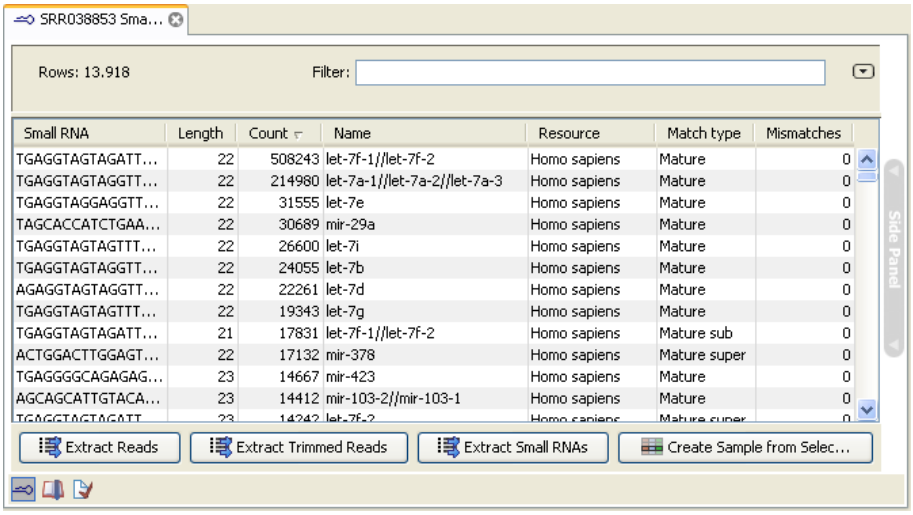


Figure 30.15: *An ungrouped annotated sample.*

Extract Trimmed Reads ( ) The same as above, except that the trimmed part has been removed.

Extract Small RNAs ( ) This will extract only one copy of each tag.

Note that for all these, you will be able to determine whether a list of DNA or RNA sequences should be produced (when working within the *Biomedical Genomics Workbench* environment, this only effects the RNA folding tools).

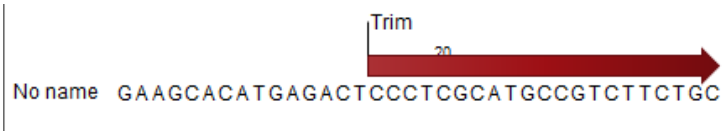


Figure 30.16: *Extracting reads from a sample.*

The button **Create Sample from Selection** ( ) can be used to create a new sample based on the tags that are selected. This can be useful in combination with filtering and sorting.

The grouped sample

An example of a grouped annotated sample is shown in figure 30.17.

The contents of the table are explained in section 30.1.3. In this section, we focus on the tools available for working with the sample.

By selecting one or more rows in the table, the buttons at the bottom of the view become active:

Open Read Mapping ( ) This will open a view showing the annotation reference sequence at the top and the tags aligned to it as shown in figure 30.18. The names of the tags indicate their status compared with the reference (e.g. Mature 5', Mature super 5', Precursor). This categorization is based on the choices you make when annotating. You can also see the annotations when using miRBase as the annotation source. In this example both the mature 5' and the mature 3' are annotated, and you can see that both are found in the sample. In the **Side Panel** to the right you can see the **Match weight** group under **Residue**

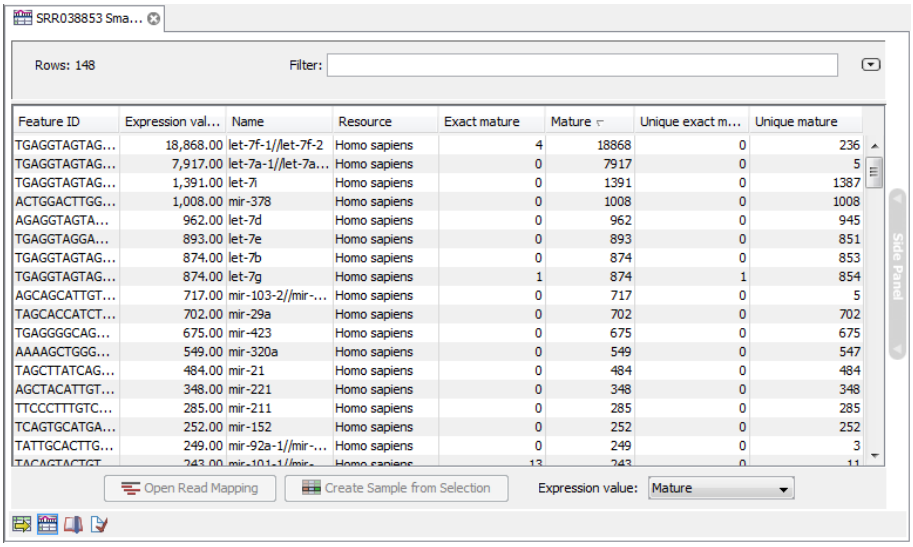


Figure 30.17: *A sample grouped on mature 5' miRNAs.*

coloring which is used to color the tags according to their relative abundance. The weight is also shown next to the name of the tag. The left side color is used for tags with low counts and the right side color is used for tags with high counts, relative to the total counts of this annotation reference. The sliders just above the gradient color box can be dragged to highlight relevant levels of abundance. The colors can be changed by clicking the box. This will show a list of gradients to choose from.

Create Sample from Selection ( ) This is used to create a new sample based on the tags that are selected. This can be useful in combination with filtering and sorting.

30.1.5 Exploring novel miRNAs

One way of doing this would be to identify interesting tags based on their counts (typically you would be interested in pursuing tags with not too low counts in order to avoid wasting efforts on tags based on reads with sequencing errors), **Extract Small RNAs** ( ) and use this list of tags as input to **Map Reads to Reference** ( ) using the genome as reference. You could then examine where the reads match, and for reads that map in otherwise unannotated regions you could select a region around the match and create a subsequence from this. The subsequence could be folded and examined to see whether the secondary structure was in agreement with the expected hairpin-type structure for miRNAs. The *Biomedical Genomics Workbench* is able to analyze expression data produced on microarray platforms and high-throughput sequencing platforms (also known as Next-Generation Sequencing platforms). The *Biomedical Genomics Workbench* provides tools for performing quality control of the data, transformation and normalization, statistical analysis to measure differential expression and annotation-based tests. A number of visualization tools such as volcano plots, MA plots, scatter plots, box plots, and heat maps are used to aid the interpretation of the results.

30.2 Experimental design

In order to make full use of the various tools for interpreting expression data, you need to know the central concepts behind the way the data is organized in the *Biomedical Genomics Workbench*.

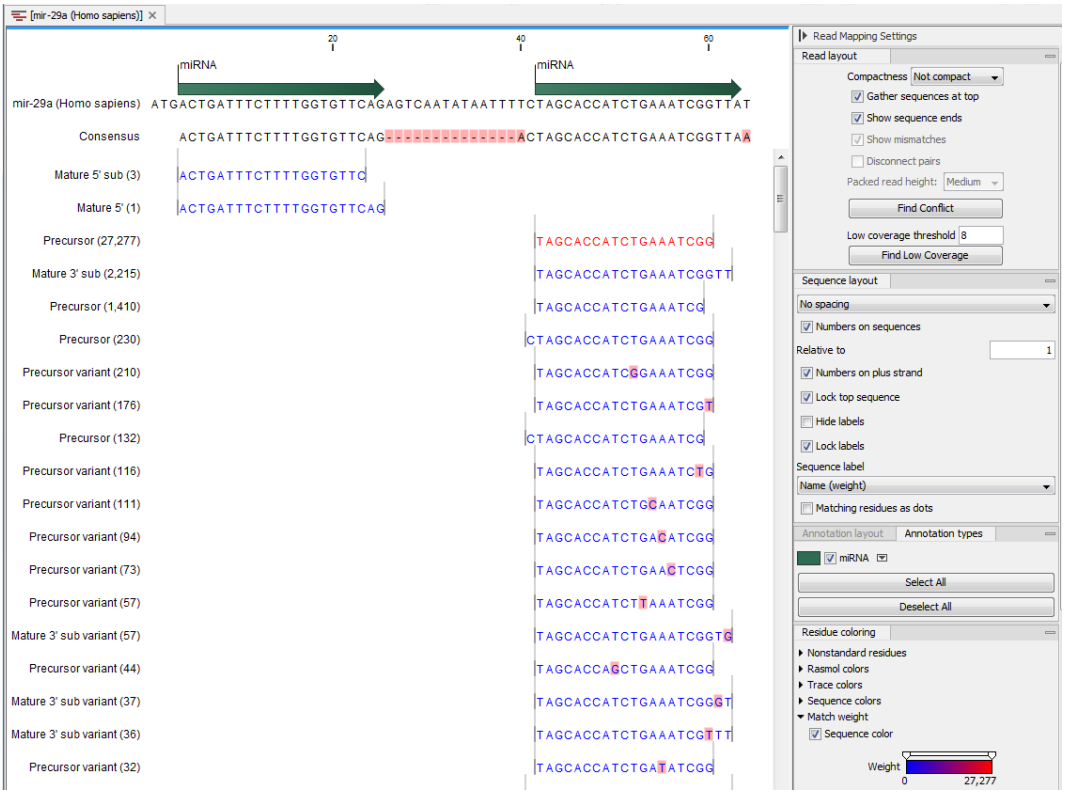


Figure 30.18: *Aligning all the variants of this miRNA from miRBase, providing a visual overview of the distribution of tags along the precursor sequence.*

The first piece of data you are faced with is the **sample**. In the Workbench, a sample contains the expression values from either one array or from sequencing data of one sample.

Note that the calculation of expression levels based on the raw sequence data is described in section 29.1 and section 30.1.

See more below on how to get your expression data into the Workbench as samples in section J.

In a sample, there is a number of **features**, usually genes, and their associated expression levels.

To analyze differential expression, you need to tell the workbench how the samples are related. This is done by setting up an **experiment**. An experiment is essentially a set of samples which are grouped. By creating an experiment defining the relationship between the samples, it becomes possible to do statistical analysis to investigate differential expression between the groups. The **Experiment** is also used to accumulate calculations like t-tests and clustering because this information is closely related to the grouping of the samples.

30.2.1 Setting up an experiment

To set up an experiment:

Toolbox | **Microarray and Small RNA Analysis (**  **)**| **Set Up Experiment (**  **)**

Select the samples that you wish to use by double-clicking or selecting and pressing the **Add** () button (see figure 30.19).

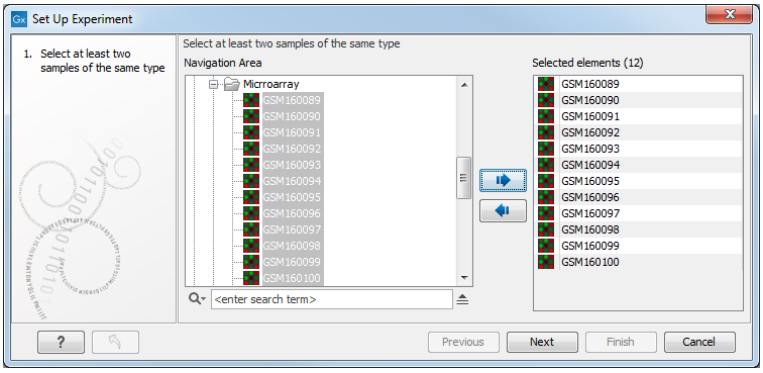


Figure 30.19: *Select the samples to use for setting up the experiment.*

Note that we use "samples" as the general term for both microarray-based sets of expression values and sequencing-based sets of expression values (e.g. an expression track from RNA-Seq).

Clicking **Next** shows the dialog in figure 30.20.

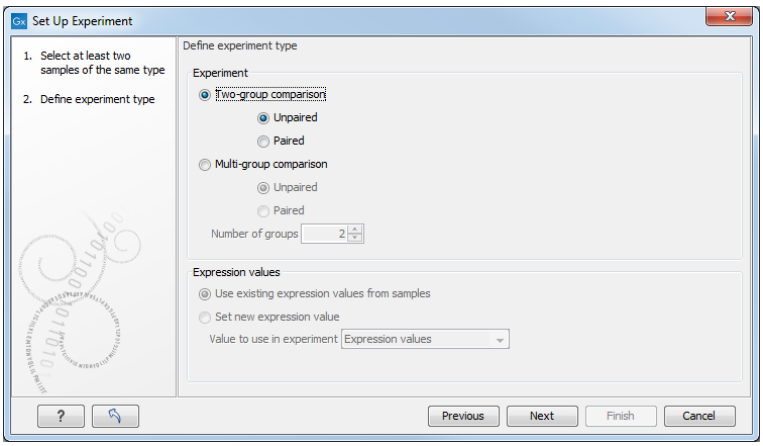


Figure 30.20: *Defining the number of groups and expression value type.*

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.** For RNA-Seq experiments, you can also choose which expression value to be used when setting up the experiment. This value will then be used for all subsequent analyses. If you choose to **Set new expression value** you can choose between the following options depending on whether you look at the gene or transcript level:

- **Genes: Unique exon reads.** The number of reads that match uniquely to the exons (including the exon-exon and exon-intron junctions).
- **Genes: Unique gene reads.** This is the number of reads that match uniquely to the gene.
- **Genes: Total exon reads.** Number of reads mapped to this gene that fall entirely within an exon or in exon-exon or exon-intron junctions. As for the "Total gene reads" this includes both uniquely mapped reads and reads with multiple matches that were assigned to an exon of this gene.
- **Genes: Total gene reads.** This is all the reads that are mapped to this gene, i.e., both reads that map uniquely to the gene and reads that matched to more positions in the reference (but fewer than the "Maximum number of hits for a read" parameter) which were assigned to this gene.
- **Genes: RPKM.** This is the expression value measured in RPKM [Mortazavi et al., 2008]: $RPKM = \frac{total\ exon\ reads}{mapped\ reads(millions) \times exon\ length\ (KB)}$. See exact definition below. Even if you have chosen the RPKM values to be used in the **Expression values** column, they will also be stored in a separate column. This is useful to store the RPKM if you switch the expression measure. See more in section 29.1.4.
- **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 30.21.

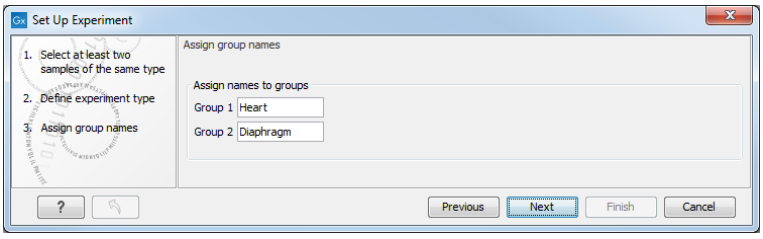


Figure 30.21: *Naming the groups.*

Depending on the number of groups selected in figure 30.20, 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 30.22.

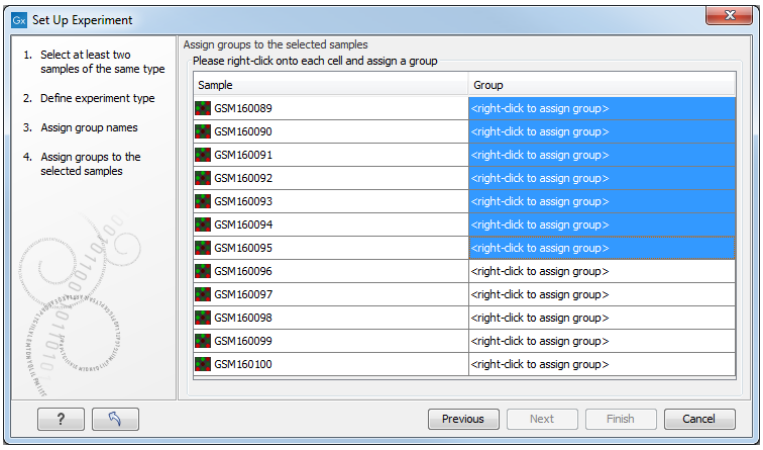


Figure 30.22: *Putting the samples into groups.*

This is where you define which group the individual sample belongs to. Simply select one or more samples (by clicking and dragging the mouse), right-click (Ctrl-click on Mac) and select the appropriate group.

Note that the samples are sorted alphabetically based on their names.

If you have chosen **Paired** in figure 30.20, 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.

30.2.2 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 30.23).

For a general introduction to table features like sorting and filtering, see section 3.3.

Unlike other tables in *Biomedical Genomics Workbench*, the experiment table has a hierarchical grouping of the columns. This is done to reflect the structure of the data in the experiment. The **Side Panel** is divided into a number of groups corresponding to the structure of the table. These are described below. Note that you can customize and save the settings of the **Side Panel** (see section 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. For more information

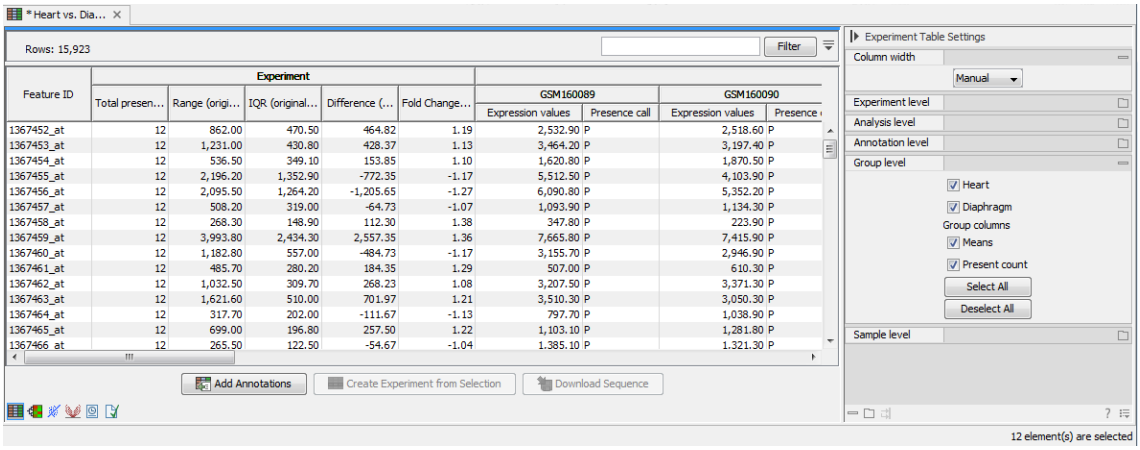


Figure 30.23: *Opening the experiment.*

on visualizing RNA-Seq read tracks from the experiment, see section 30.3.4.

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 30.24).

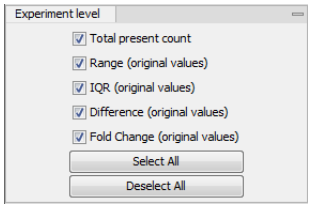


Figure 30.24: *The initial view of the experiment level for a two-group experiment.*

Initially, it has one header for the whole **Experiment**:

- **Range (original values)**. The 'Range' column contains the difference between the highest and the lowest expression value for the feature over all the samples. If a feature has the value NaN in one or more of the samples the range value is NaN.
- **IQR (original values)**. The 'IQR' column contains the inter-quantile range of the values for a feature across the samples, that is, the difference between the 75 %-ile value and the 25

%-ile value. For the IQR values, only the numeric values are considered when percentiles are calculated (that is, NaN and +Inf or -Inf values are ignored), and if there are fewer than four samples with numeric values for a feature, the IQR is set to be the difference between the highest and lowest of these.

- **Difference (original values)**. For a two-group experiment the 'Difference' column contains the difference between the mean of the expression values across the samples assigned to group 2 and the mean of the expression values across the samples assigned to group 1. Thus, if the mean expression level in group 2 is higher than that of group 1 the 'Difference' is positive, and if it is lower the 'Difference' is negative. For experiments with more than two groups the 'Difference' contains the difference between the maximum and minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs before the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

- **Fold Change (original values)**. For a two-group experiment the 'Fold Change' tells you how many times bigger the mean expression value in group 2 is relative to that of group 1. If the mean expression value in group 2 is bigger than that in group 1 this value is the mean expression value in group 2 divided by that in group 1. If the mean expression value in group 2 is smaller than that in group 1 the fold change is the mean expression value in group 1 divided by that in group 2 with a negative sign. Thus, if the mean expression levels in group 1 and group 2 are 10 and 50 respectively, the fold change is 5, and if the and if the mean expression levels in group 1 and group 2 are 50 and 10 respectively, the fold change is -5. Entries of plus or minus infinity in the 'Fold Change' columns of the Experiment area represent those where one of the expression values in the calculation is a 0. For experiments with more than two groups, the 'Fold Change' column contains the ratio of the maximum of the mean expression values of the groups to the minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs before the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

Thus, the sign of the values in the 'Difference' and 'Fold change' columns give the direction of the trend across the groups, going from group 1 to group 2, etc.

If the samples used are Affymetrix GeneChips samples and have 'Present calls' there will also be a 'Total present count' column containing the number of present calls for all samples.

The columns under the 'Experiment' header are useful for filtering purposes, e.g. you may wish to ignore features that differ too little in expression levels to be confirmed e.g. by qPCR by filtering on the values in the 'Difference', 'IQR' or 'Fold Change' columns or you may wish to ignore features that do not differ at all by filtering on the 'Range' column.

If you have performed normalization or transformation (see sections 30.4.3 and 30.4.2, respectively), the IQR of the normalized and transformed values will also appear. Also, if you later choose to transform or normalize your experiment, columns will be added for the transformed or normalized values.

Note! It is very common to filter features on fold change values in expression analysis and fold change values are also used in volcano plots, see section 30.6.5. There are different definitions of 'Fold Change' in the literature. The definition that is used typically depends on the original scale of the data that is analyzed. For data whose original scale is *not* the log scale the standard

definition is the ratio of the group means [Tusher et al., 2001]. This is the value you find in the 'Fold Change' column of the experiment. However, for data whose original *is* the log scale, the difference of the mean expression levels is sometimes referred to as the fold change [Guo et al., 2006], and if you want to filter on fold change for these data you should filter on the values in the 'Difference' column. Your data's original scale will e.g. be the log scale if you have imported Affymetrix expression values which have been created by running the RMA algorithm on the probe-intensities.

Analysis level

The results of each statistical test performed are in the columns listed in this area. In the table, a heading is given for each test. Information about the results of statistical tests are described in the statistical analysis section (see section 30.6).

An example of Analysis level settings is shown in figure 30.25.

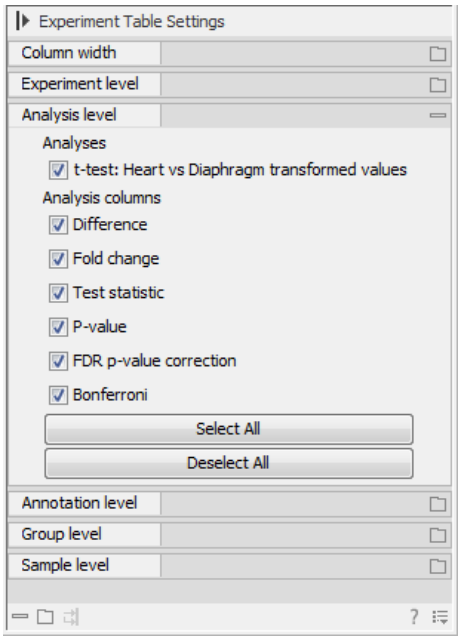


Figure 30.25: *An example of columns available under the Analysis level section.*

Note: Some column names here are the same as ones under the Experiment level, but the results here are from statistical tests, while those under the Experiment level section are calculations carried out directly on the expression levels.

Annotation level

If your experiment is annotated (see section 30.2.3), the annotations will be listed in the **Annotation level** group as shown in figure 30.26.

In order to avoid too much detail and cluttering the table, only a few of the columns are shown per default.

Note that if you wish a different set of annotations to be displayed each time you open an experiment, you need to save the settings of the **Side Panel** (see section 4.6).

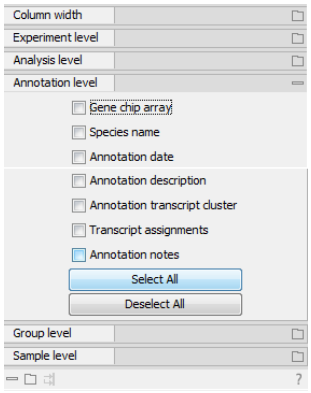


Figure 30.26: *An annotated experiment.*

Group level

At the group level, you can show/hide entire groups (*Heart* and *Diaphragm* in figure 30.23). 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 30.4.3 and 30.4.2, respectively), the means of the normalized and transformed values will also appear.

An example is shown in figure 30.27.

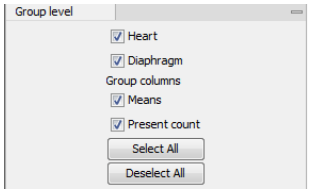


Figure 30.27: *Group level .*

Sample level

In this part of the side panel, you can control which columns to be displayed for each sample. Initially this is the all the columns in the samples.

If you have performed normalization or transformation (see sections 30.4.3 and 30.4.2, respectively), the normalized and transformed values will also appear.

An example is shown in figure 30.28.

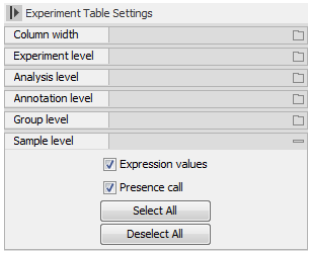


Figure 30.28: *Sample level when transformation and normalization has been performed.*

Creating a sub-experiment from a selection

If you have identified a list of genes that you believe are differentially expressed, you can create a subset of the experiment. (Note that the filtering and sorting may come in handy in this situation, see section 3.3).

To create a sub-experiment, first select the relevant features (rows). If you have applied a filter and wish to select all the visible features, press Ctrl + A (⌘ + A on Mac). Next, press the **Create Experiment from Selection** ( ) button at the bottom of the table (see figure 30.29).

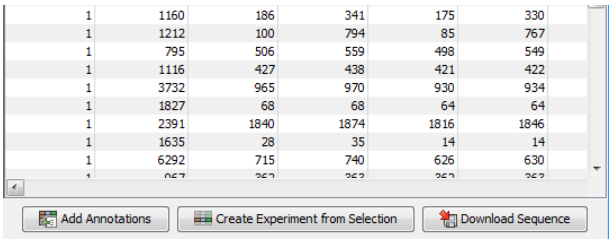


Figure 30.29: *Create a subset of the experiment by clicking the button at the bottom of the experiment table.*

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 30.30).

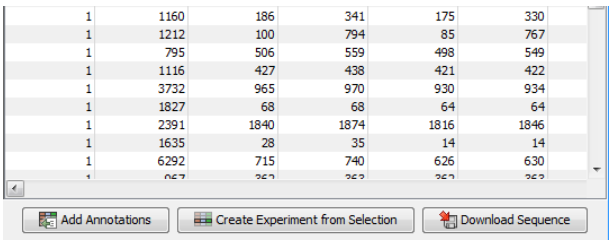


Figure 30.30: *Select sequences and press the download button.*

This will open a dialog where you specify where the sequences should be saved. You can learn more about opening and viewing sequences in chapter 10. You can now use the downloaded sequences for further analysis in the Workbench.

30.2.3 Adding annotations to an experiment

Annotation files provide additional information about each feature. This information could be which GO categories the protein belongs to, which pathways, various transcript and protein identifiers etc. See section J for information about the different annotation file formats that are supported *Biomedical Genomics Workbench*.

The annotation file can be imported into the Workbench and will get a special icon ( ). See an overview of annotation formats supported by *Biomedical Genomics Workbench* in section J. In

order to associate an annotation file with an experiment, either select the annotation file when you set up the experiment (see section 30.2.1), or click:

> **Toolbox** | **Microarray and Small RNA 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 30.2.2. You can also add annotations by pressing the **Add Annotations** () button at the bottom of the table (see figure 30.31).

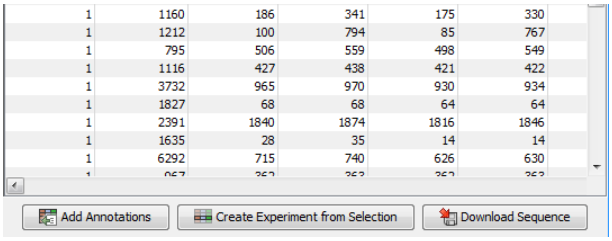


Figure 30.31: *Adding annotations by clicking the button at the bottom of the experiment table.*

This will bring up a dialog where you can select the annotation file that you have imported together with the experiment you wish to annotate. Click **Next** to specify settings as shown in figure 30.32).

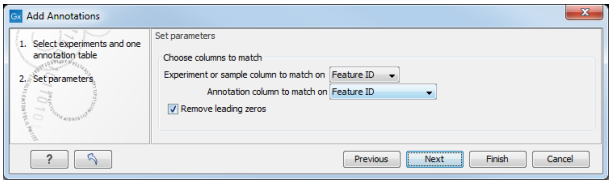


Figure 30.32: *Choosing how to match annotations with samples.*

In this dialog, you can specify how to match the annotations to the features in the sample. The Workbench looks at the columns in the annotation file and lets you choose which column that should be used for matching to the feature IDs in the experimental data (experiment or sample) as well as for the annotations. Usually the default is right, but for some annotation files, you need to select another column.

Some annotation files have leading zeros in the identifier which you can remove by checking the **Remove leading zeros** box.

Note! Existing annotations on the experiment will be overwritten.

30.2.4 Scatter plot view of an experiment

At the bottom of the experiment table, you can switch between different views of the experiment (see figure 30.33).

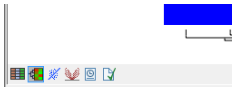


Figure 30.33: *An experiment can be viewed in several ways.*

One of the views is the **Scatter Plot** (). The scatter plot can be adjusted to show e.g. the group means for two groups (see more about how to adjust this below).

An example of a scatter plot is shown in figure 30.34.

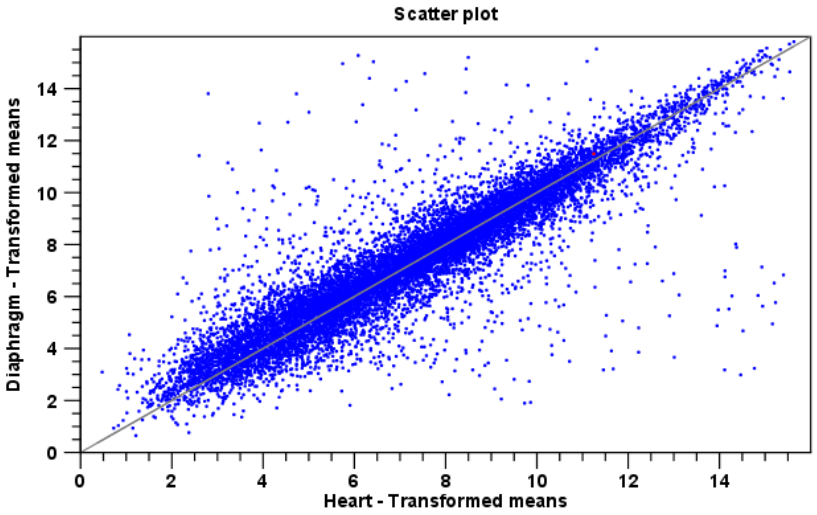


Figure 30.34: *A scatter plot of group means for two groups (transformed expression values).*

In the **Side Panel** to the left, there are a number of options to adjust this view. Under **Graph preferences**, you can adjust the general properties of the scatter plot:

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Draw x = y axis**. This will draw a diagonal line across the plot. This line is shown per default.
- **Line width**

  - Thin
  - Medium
  - Wide

- **Line type**
  - None
  - Line
  - Long dash
  - Short dash

- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

- Show Pearson correlation When checked, the Pearson correlation coefficient (r) is displayed on the plot.

Below the general preferences, you find the **Dot properties** preferences, where you can adjust coloring and appearance of the dots:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Finally, the group at the bottom - **Values to plot** - is where you choose the values to be displayed in the graph. The default for a two-group experiment is to plot the group means.

Note that if you wish to use the same settings next time you open a scatter plot, you need to save the settings of the **Side Panel** (see section 4.6).

30.2.5 Cross-view selections

There are a number of different ways of looking at an experiment as shown in figure 30.35).

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

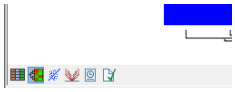


Figure 30.35: *An experiment can be viewed in several ways.*

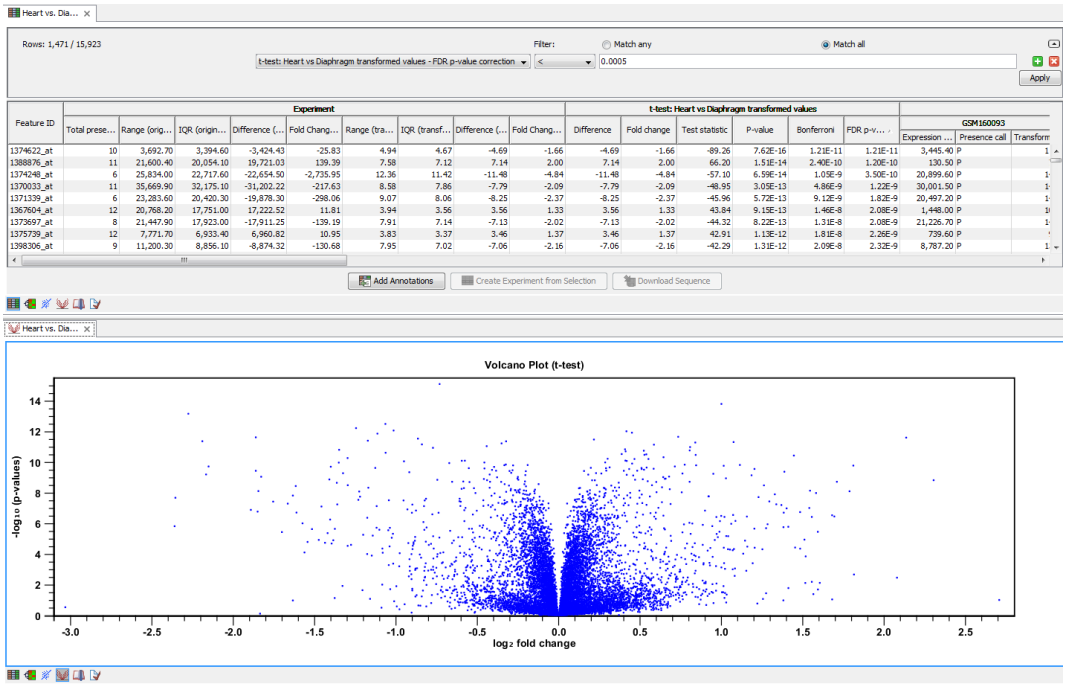


Figure 30.36: *A split view showing an experiment table at the top and a volcano plot at the bottom (note that you need to perform statistical analysis to show a volcano plot, see section 30.6).*

while you click one of the view buttons in figure 30.35, 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 30.36.

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 30.2.2) you typically want to choose 'Difference'.

30.3 Working with tracks and experiments

The *Biomedical Genomics Workbench* provides several tools for the analysis, organization, and visualization of expression data. In this section, we describe how Tracks and Experiments complement each other, and how they can be used together for the analysis of transcriptomics data.

30.3.1 Data structures for transcriptomics

The two main data structures used for transcriptomics data analysis in the *Biomedical Genomics Workbench* are tracks and experiments.

Tracks, also known as 'Genome Browser View', (see section 19) are the fundamental building blocks for data analysis in the *Biomedical Genomics Workbench*, where all information is tied to genomic positions. A central coordinate-system is provided by a reference genome, which allows different types of data or results for different samples to be seen and analyzed together.

Experiments (see section 30.2), on the other hand, are used to represent complex relationships between expression samples, and to carry out statistical analysis (see section 30.6) of differential expression.

Tracks and experiments are intimately related, and it is possible in most cases to convert from one type to the other.

From Tracks to Experiments

When carrying out RNA-seq analysis using the **RNA-Seq Analysis** tool, the starting point is a set of reads from a sequencing study. As part of the RNA-Seq Analysis tool, these reads are mapped onto a reference genome. The RNA-Seq tool produces expression tracks, which are compatible with the reference genome, and can be visualized together with the genome in the

Genome Browser View (see section 19).

You can find more information about the RNA-Seq Analysis tool in section 29.1).

Once expression tracks have been obtained from the RNA-Seq Analysis tool, they can be used as sequencing-based sets of expression values in setting up an experiment. This can be done using the **Set up Experiment** tool and is described in more detail in section 30.2.

An experiment set up in this manner from expression tracks is intimately coupled to the tracks it originated from. To see this coupling in action, perform the following steps:

1. Use the **Set up Experiment** tool on two or more expression tracks to set up an experiment, as described in section 30.2.
2. Save and open the resulting experiment, by double-clicking its name in the **Navigation Area**.
3. Use the **Create New Genome Browser View** tool to create a new Genome Browser View from the expression tracks you used to set up the experiment, as described in section 19.
4. Save and open the resulting Genome Browser View by double-clicking its name in the **Navigation Area**.
5. Drag the experiment tab downwards, until you see the blue shadow indicating the resulting placement (figure 30.37), and drop it in place. You should now have a divided view, with

the experiment in the bottom half (figure 30.38).

6. Clicking on any line in the experiment will now automatically jump to the corresponding genomic location in the upper view. Use the **Zoom to Selection** ( ) button to zoom in to the desired genomic region.

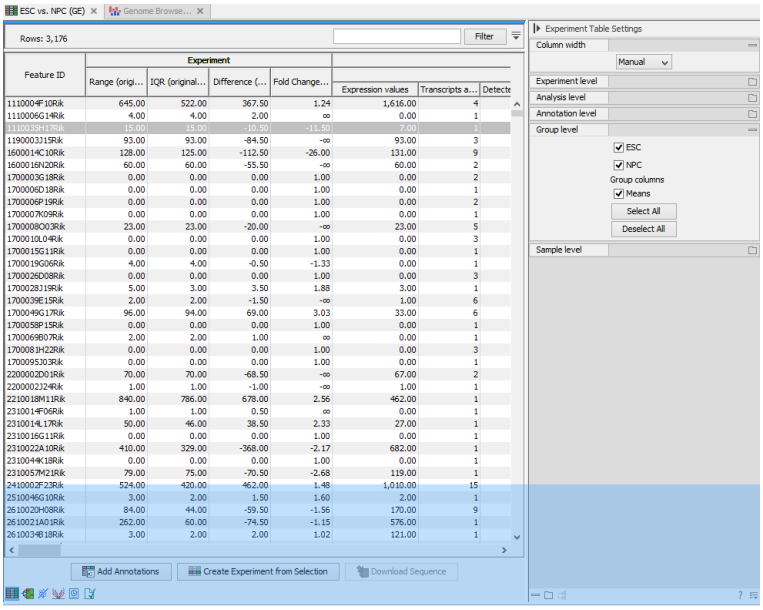


Figure 30.37: *Dragging a tab to the lower half of the view area.*

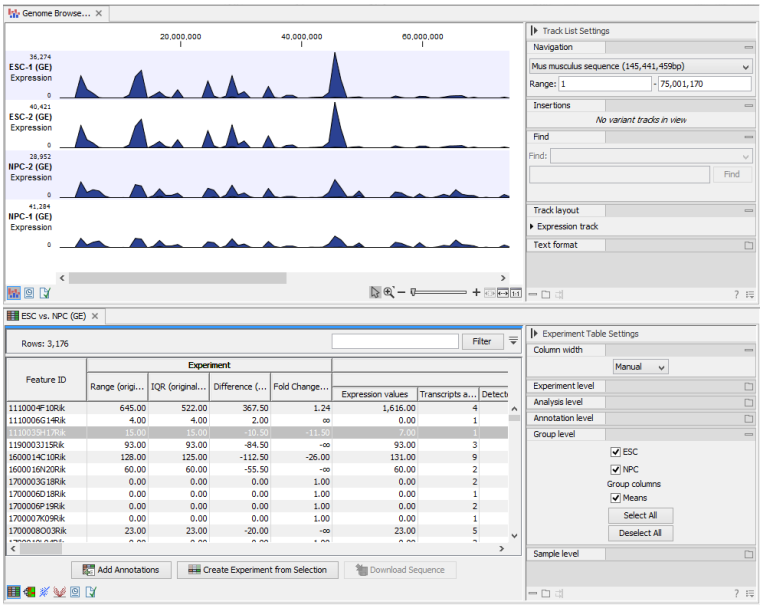


Figure 30.38: *After dropping a tab to the lower half othe view area.*

From Experiments to Tracks

Experiments can be used to carry out statistical analysis on the expression values obtained from RNA-seq analysis as described in section 30.6. The results of the statistical analysis are annotated on the experiment as additional columns.

It can be advantageous to visualize the results of the statistical analysis as tracks. The **Extract Differentially Expressed Genes** tool in the *Biomedical Genomics Workbench* enables the

conversion of experiments to tracks. You can find the **Extract Differentially Expressed Genes** tool here:

Toolbox | **Microarray and Small RNA** | **Extract Differentially Expressed Genes** () .

30.3.2 Running the Extract Differentially Expressed Genes tool

After you start the tool, you are presented with a wizard where you can choose the experiment that you would like to create a track of. The **Extract Differentially Expressed Genes** tool can be run on experiments with associated genomic information, such as those created using expression tracks from the **RNA-Seq Analysis** tool.

In the case where the experiment has associated genomic information, the **Extract Differentially Expressed Genes** tool will automatically infer these and the wizard will jump directly to the filtering step, as shown in figure 30.40.

In the case where the experiment does not have associated genomic information, you will first need to specify how the genomic information should be obtained in the Parameters step of the **Extract Differentially Expressed Genes** tool (figure 30.41).

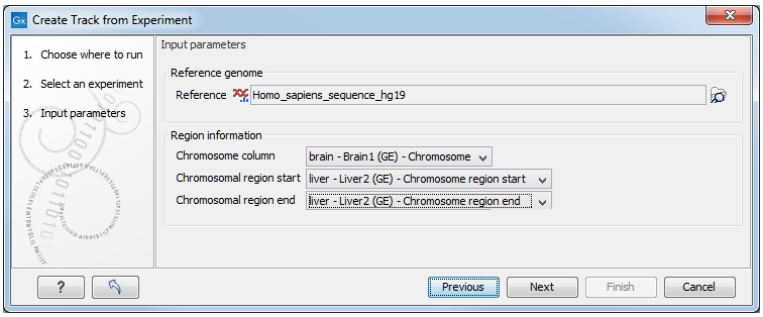


Figure 30.39: *The "Input parameters" step in the Extract Differentially Expressed Genes tool.*

In the Input parameters step, you must specify the following parameters:

- **Reference genome.** The chosen genome will be used as the reference genome for the resulting track.
- **Chromosome column.** The column containing the chromosome names must be chosen from the drop-down menu.
- **Chromosomal region start.** The column containing the start of the genomic regions must be chosen from the drop-down menu.
- **Chromosomal region end.** The column containing the end of the genomic regions must be chosen from the drop-down menu.

Note! The drop-down menus will only contain the columns that potentially represent the information required by the given parameter. If the experiment does not contain any columns that potentially represent the required genomic information, the drop-down menus may appear empty. In this case, it is not possible to convert the given experiment to a track.

In the Filtering step (figure 30.40), you have the following options:

- **Filter based on statistical analysis results** This allows to filter which annotations are transferred to the track on the basis of the statistical analysis. To enable filtering, check the **Filter based on statistical analysis results** checkbox. The filtering option is only available if a statistical analysis has previously been carried out on the Experiment, and the drop-down menu will only contain the statistical analyses that are present on the Experiment.
- **Statistical analysis** Allows you to choose statistical analysis from the drop-down list. The selection of available statistical analyses depends on which tests have been used when you set up the experiment that you are about to convert to track format.
- **Type of p-value** This drop-down menu allows you to select between raw and corrected p-values (see section 30.6.4). Only the types of p-values available for the given statistical analysis will be present in the drop-down menu.
- **Maximum p-value** In this input field, you can enter the maximum allowed p-value, as a number between 0 and 1. If you do not want any filtering based on p-value, enter 1.
- **Minimum fold-change value** You can also specify the minimum allowed fold-change value as a number greater than zero. If you do not want any filtering based on fold-change, enter 0.

You can then select in the drop-down menu which analysis you want to use for filtering.

The fold change values are stored as different columns in the experiment, depending on the statistical analysis performed. The Extract Differentially Expressed Genes tool will automatically use the fold-change column appropriate for the different statistical analyses:

- Kal's Z-test (see section 30.6.2): Proportions fold change.
- Baggerley's test(see section 30.6.2): Weighted proportions fold change.
- T-test (see section 30.6.3): Fold change.
- ANOVA (see section 30.6.3): Max fold change.
- Empirical analysis of DGE (see section 30.6.1): Fold change.

The resulting track will contain only differentially expressed genes whose p-value is lower than the specified threshold and whose fold-enrichment is above the specified threshold.

If the chosen statistical analysis was performed on several pairs of groups, there will be an output track for each tested pair of groups. For example, if the same statistical analysis has been carried out on 'group 1 vs. group 2' and 'group 1 vs. group 3', then the output will contain two tracks, where one is filtered according to the 'group 1 vs. group 2' analysis results and the other one is filtered according to the 'group 1 vs. group 3' analysis results.

When running the **Extract Differentially Expressed Genes** tool as part of a workflow, there are a few differences in how the parameters are set (see figure 30.42).

- The **Source of genomic information** parameter determines the behavior of the algorithm if the incoming experiment is *not* coupled to a genome. If the value of this parameter

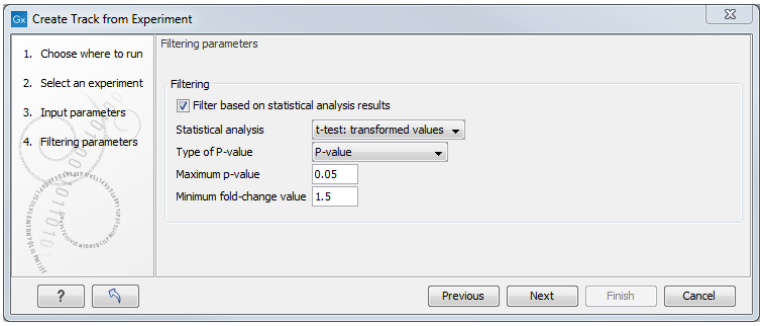


Figure 30.40: *The filtering step in the Extract Differentially Expressed Genes tool.*

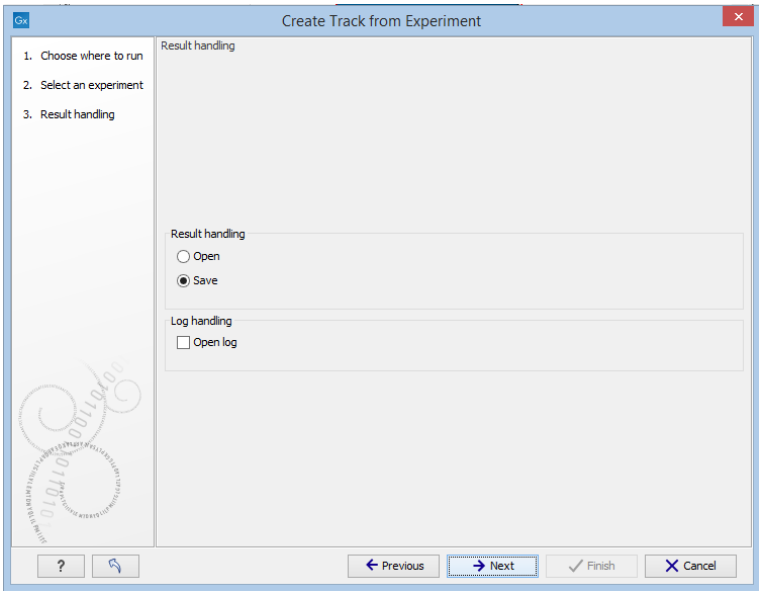


Figure 30.41: *The result handling step in the Extract Differentially Expressed Genes tool.*

is set to **Require genomic information in experiment**, then the algorithm will expect the incoming experiment to be coupled to a genome, and will fail with an error alerting the user in case the experiment does not fulfill this criterion. If the value of the parameter is set to **Automatic: use genomic information if available**, then the algorithm will still use the genomic information in a genome-coupled experiment. But if this information is not available, the algorithm will attempt to use the information specified by the user in the workflow parameters. *Note:* If the incoming experiment *is* coupled to a genome (as will usually be the case), the value of this parameter makes no difference.

- In a workflow setting, the column titles for the chromosome, region end and region start fields can be specified as texts. These fields may be left empty, if the incoming experiment contains the genomic information. If filling out these fields, note that the format for this text is very strict, and must exactly match the text appearing in the drop-down menu when running the tool from the toolbox. For example, if 'Chromosome' is a sample-specific column, for a sample called 'Liver (GE)' in the 'liver' group in the experiment, then the column name text will be: 'liver - Liver (GE) - Chromosome'.

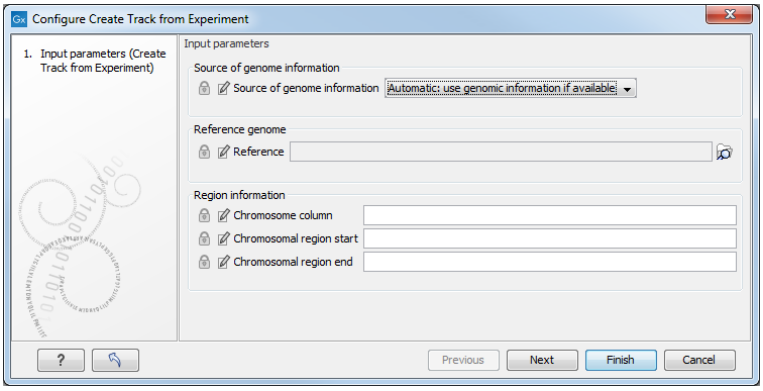


Figure 30.42: *Setting the parameters for the Extract Differentially Expressed Genes tool in a workflow*

30.3.3 Interpreting the results of the Extract Differentially Expressed Genes tool

The **Extract Differentially Expressed Genes** tool will produce a track or several tracks, if filtering based on analysis results was chosen. The track(s) will contain the following annotations:

- All experiment-specific columns from the experiment
- All user-defined annotations added to the experiment
- All analysis-specific columns from the experiment
- All group-specific columns from the experiment
- Those of the following sample-specific columns when present in the experiment (for each sample): Expression values, Total exon reads, and RPKM.

Two different view options exist: the Genome Browser View and the Table View. When opening the annotated output result, the default view is the Genome Browser View. It is possible to open both views in split view by holding down the Ctrl key while clicking on the table icon in the lower left corner of the View Area. The two different views are linked together. This means that when you click once on an entry in the table, the Genome Browser View will jump the selected region. With the **Zoom to Selection** ( ) button it is possible to jump to and zoom in on the selected region (figure 30.43).

The results of any statistical test executed on the experiment, including fold-changes and p-values, can be seen in the tooltip when hovering over each region in the annotation track shown in Genome Browser View (figure 30.44).

30.3.4 Visualizing RNA-Seq read tracks for the experiment

When working with RNA-Seq data, the experiment can be used to browse the read mappings to investigate how the reads supporting each sample are mapped. This is done by creating a track list:

File | **New** | **Genome Browser View ( )**

Select the mapping and expression tracks of the samples you wish to visualize together and select any annotation tracks (e.g. gene and mRNA) to be included for visualization **Finish**.

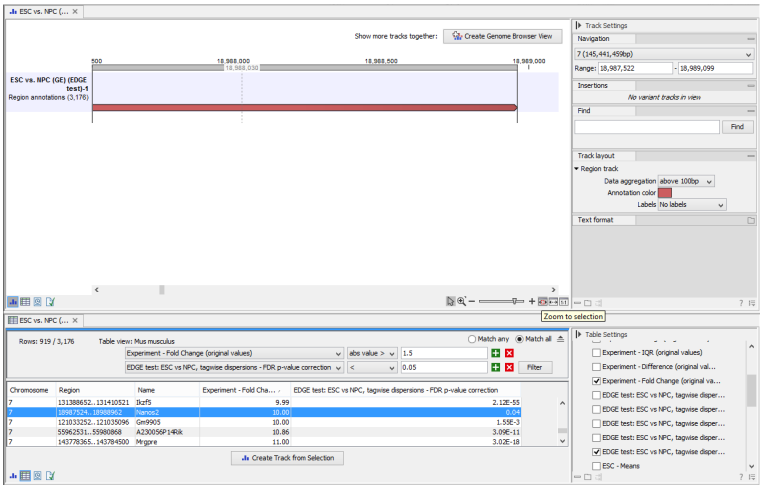


Figure 30.43: *Viewing the track produced by the Extract Differentially Expressed Genes Tool*

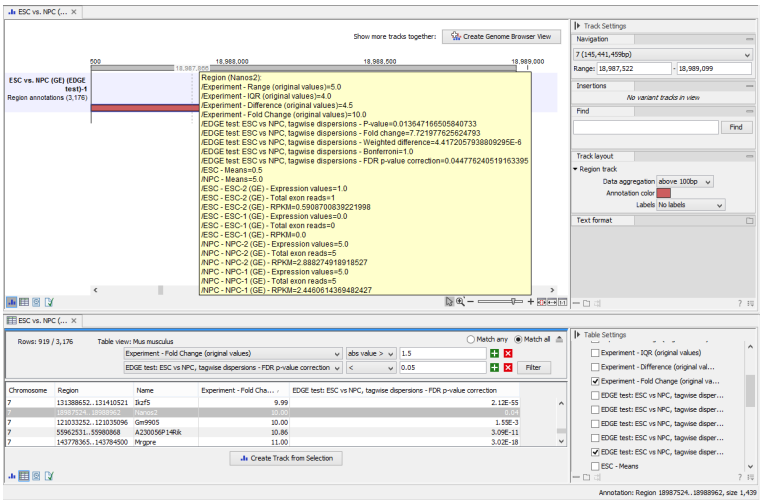


Figure 30.44: *The annotations on the track produced by the Extract Differentially Expressed Genes Tool*

Once the track list is shown, create a split view or drag the tab of the view on to a second screen (if you have two screens). Clicking a row in the table makes the track list view jump to that location, allowing for quick inspection of interesting parts of the RNA-Seq read mapping (see an example in figure 30.45. Note that the **Zoom to selection** ( ) button can be used to adjust the zoom level to fit the region selection.

Please note that at least one of the expression tracks used in the experiment have to be included in the track list in order for the link between the two to work.

30.4 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

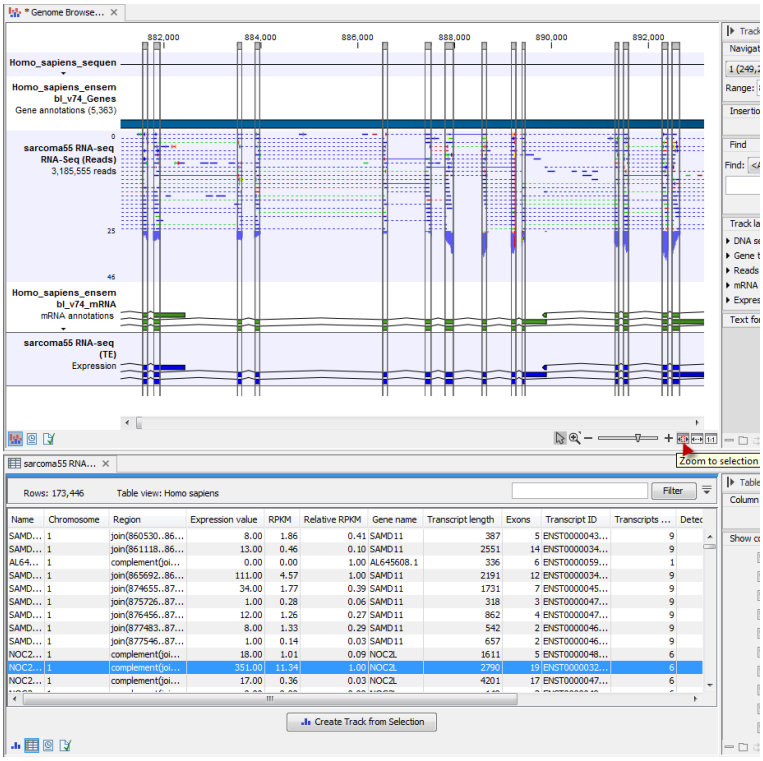


Figure 30.45: *RNA-Seq results shown in a split view with an experiment table at the bottom and a track list with read mappings of several samples at the top.*

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).

30.4.1 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 30.46).

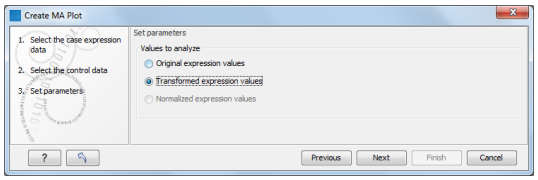


Figure 30.46: *Selecting which version of the expression values to analyze. In this case, the values have not been normalized, so it is not possible to select normalized values.*

In this case, the values have not been normalized, so it is not possible to select normalized values.

30.4.2 Transformation

The *Biomedical Genomics Workbench* lets you transform expression values based on logarithm and adding a constant:

> **Toolbox** | **Microarray and Small RNA Analysis (**)| **Transformation and Normalization** | **Transform (**)

Select a number of samples (() or ()) or an experiment () and click **Next**.

This will display a dialog as shown in figure 30.47.

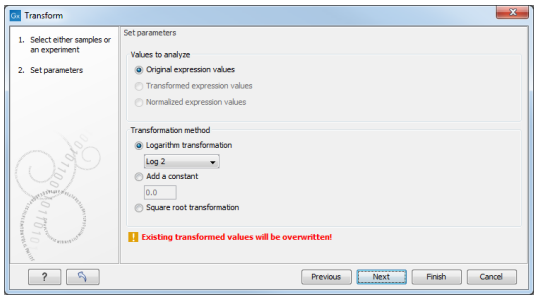


Figure 30.47: *Transforming expression values.*

At the top, you can select which values to transform (see section 30.4.1).

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.

30.4.3 Normalization

The *Biomedical Genomics Workbench* lets you normalize expression values.

To start the normalization:

> **Toolbox** | **Microarray and Small RNA 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 30.48.

At the top, you can choose three kinds of normalization (for mathematical descriptions see [Bolstad et al., 2003]):

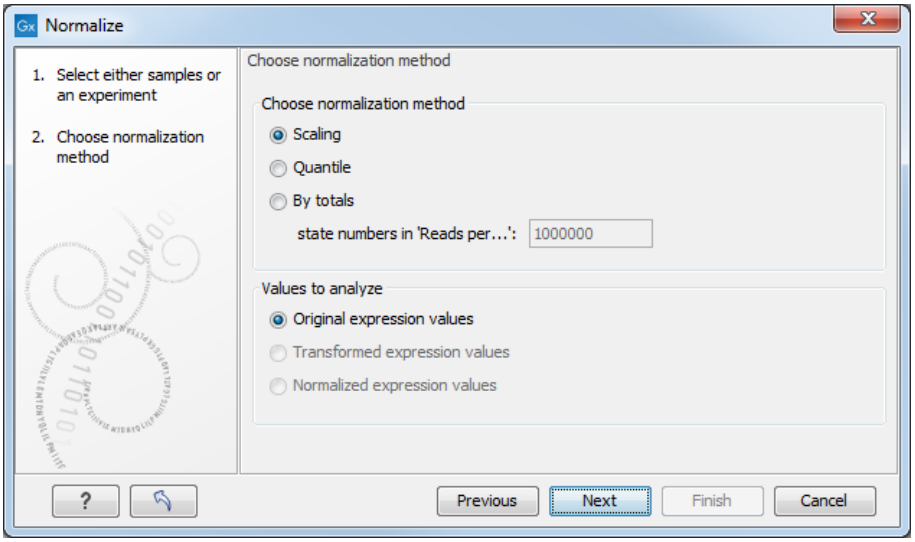


Figure 30.48: *Choosing normalization method.*

- **Scaling**. The sets of the expression values for the samples will be multiplied by a constant so that the sets of normalized values for the samples have the same 'target' value (see description of the **Normalization value** below).
- **Quantile**. The empirical distributions of the sets of expression values for the samples are used to calculate a common target distribution, which is used to calculate normalized sets of expression values for the samples.
- **By totals**. This option is intended to be used with count-based data, i.e. data from RNA-seq, small RNA or expression profiling by tags. A sum is calculated for the expression values in a sample. The transformed value are generated by dividing the input values by the sample sum and multiplying by the factor (e.g. per '1,000,000').

Figures 30.49 and 30.50 show the effect on the distribution of expression values when using scaling or quantile normalization, respectively.

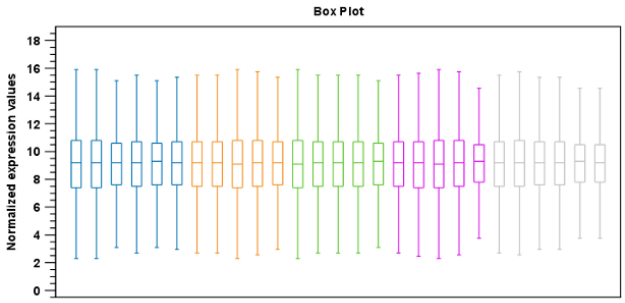


Figure 30.49: *Box plot after scaling normalization.*

At the bottom of the dialog in figure 30.48, you can select which values to normalize (see section 30.4.1).

Clicking **Next** will display a dialog as shown in figure 30.51.

The following parameters can be set:

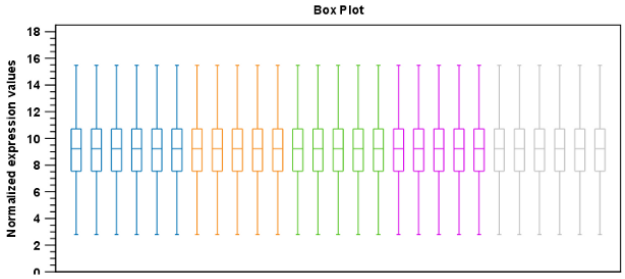


Figure 30.50: *Box plot after quantile normalization.*

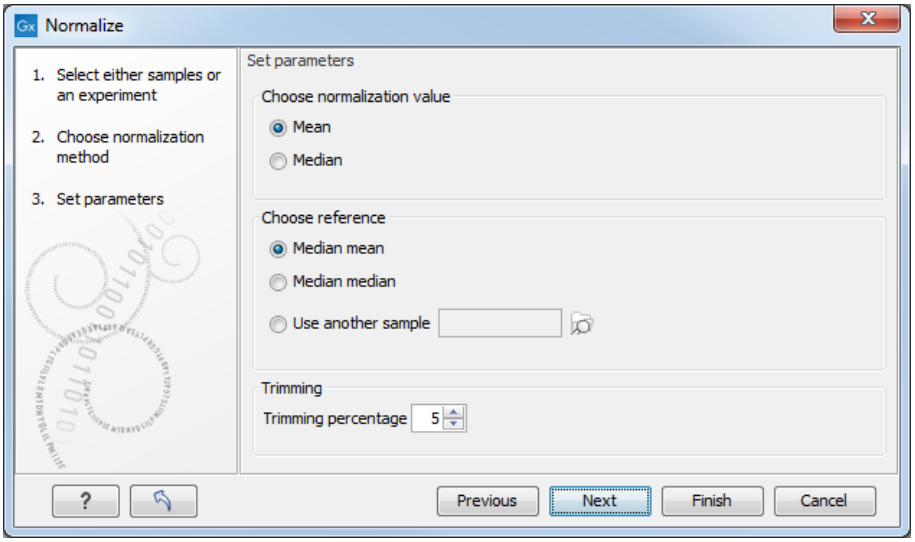


Figure 30.51: *Normalization settings.*

- **Normalization value**. The type of value of the samples which you want to ensure are equal for the normalized expression values
  - **Mean**.
  - **Median**.
- **Reference**. The specific value that you want the normalized value to be after normalization.
  - **Median mean**.
  - **Median median**.
  - **Use another sample**.
- **Trimming percentage**. Expression values that lie below the value of this percentile, or above 100 minus the value of this percentile, in the empirical distribution of the expression values in a sample will be excluded when calculating the normalization and reference values.

Click **Finish** to start the tool.

30.5 Quality control

The *Biomedical Genomics Workbench* includes a number of tools for quality control. These allow visual inspection of the overall distributions, variability and similarity of the sets of expression

values in samples, and may be used to spot unwanted systematic differences between samples, outlying samples and samples of poor quality, that you may want to exclude.

30.5.1 Creating box plots - analyzing distributions

In most cases you expect the majority of genes to behave similarly under the conditions considered, and only a smaller proportion to behave differently. Thus, at an overall level you would expect the distributions of the sets of expression values in samples in a study to be similar. A boxplot provides a visual presentation of the distributions of expression values in samples. For each sample the distribution of it's values is presented by a line representing a center, a box representing the middle part, and whiskers representing the tails of the distribution. Differences in the overall distributions of the samples in a study may indicate that normalization is required before the samples are comparable. An atypical distribution for a single sample (or a few samples), relative to the remaining samples in a study, could be due to imperfections in the preparation and processing of the sample, and may lead you to reconsider using the sample(s).

To create a box plot:

> **Toolbox** | **Microarray and Small RNA Analysis** ( )| **Quality Control** | **Create Box Plot** ( )

Select a number of samples (( ) or ( )) or an experiment ( ) and click **Next**.

This will display a dialog as shown in figure 30.52.

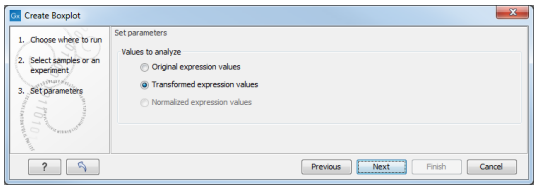


Figure 30.52: *Choosing values to analyze for the box plot.*

Here you select which values to use in the box plot (see section 30.4.1).

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 30.53.

Note that the boxes per default are colored according to their group relationship. At the bottom you find the names of the samples, and the y-axis shows the expression values (note that sample names are not shown in figure 30.53).

Per default the box includes the IQR values (from the lower to the upper quartile), the median is displayed as a line in the box, and the whiskers extend 1.5 times the height of the box.

In the **Side Panel** to the left, there is a number of options to adjust this view. Under **Graph preferences**, you can adjust the general properties of the box plot (see figure 30.54).

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.

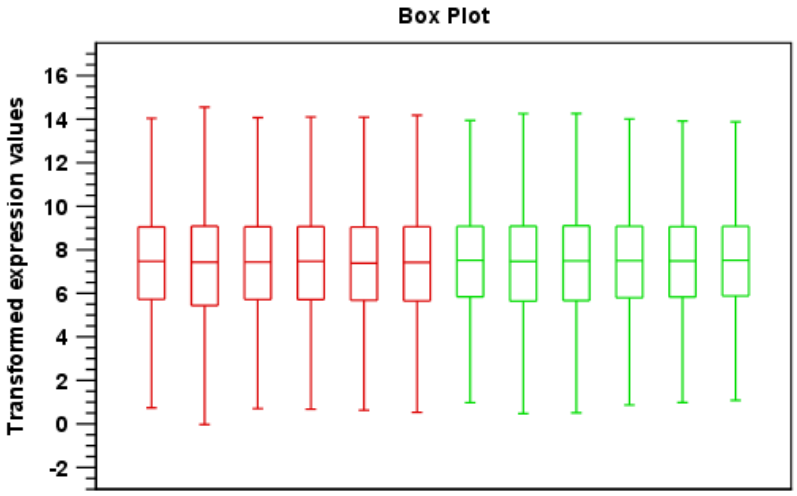


Figure 30.53: *A box plot of 12 samples in a two-group experiment, colored by group.*

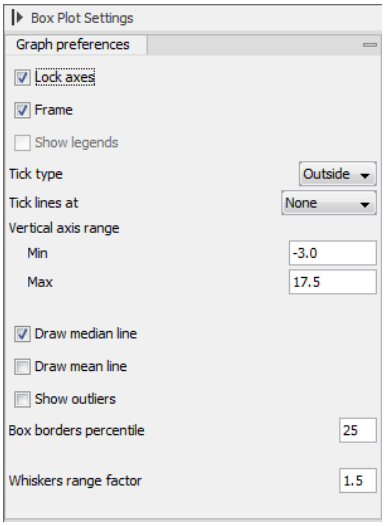


Figure 30.54: *Graph preferences for a box plot.*

- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

- **Draw median line**. This is the default - the median is drawn as a line in the box.
- **Draw mean line**. Alternatively, you can also display the mean value as a line.
- **Show outliers**. The values outside the whiskers range are called outliers. Per default they are not shown. Note that the dot type that can be set below only takes effect when outliers are shown. When you select and deselect the **Show outliers**, the vertical axis range is automatically re-calculated to accommodate the new values.

Below the general preferences, you find the **Lines and dots** preferences, where you can adjust coloring and appearance (see figure 30.55).

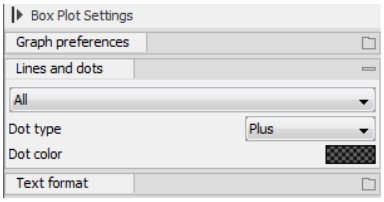


Figure 30.55: *Lines and dot preferences for a box plot.*

- **Select sample or group**. When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.
- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a box plot, you need to save the settings of the **Side Panel** (see section 4.6).

Interpreting the box plot

This section will show how to interpret a box plot through a few examples.

First, if you look at figure 30.56, you can see a box plot for an experiment with 5 groups and 27 samples.

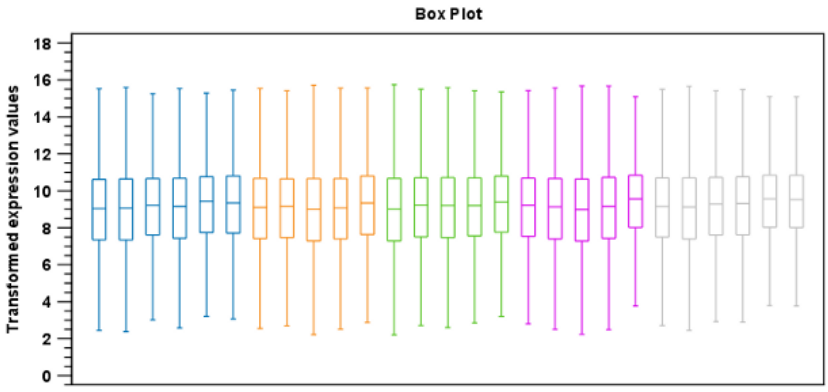


Figure 30.56: *Box plot for an experiment with 5 groups and 27 samples.*

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 30.57 shows a box plot for the same experiment after quantile normalization: the distributions have been brought into par.

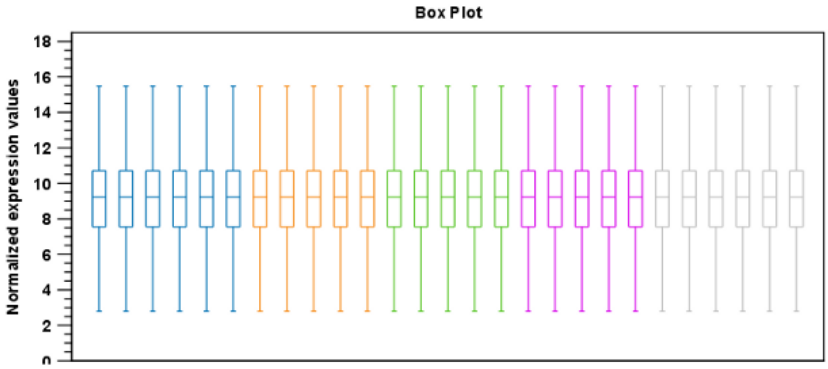


Figure 30.57: *Box plot after quantile normalization.*

In figure 30.58 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.

30.5.2 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

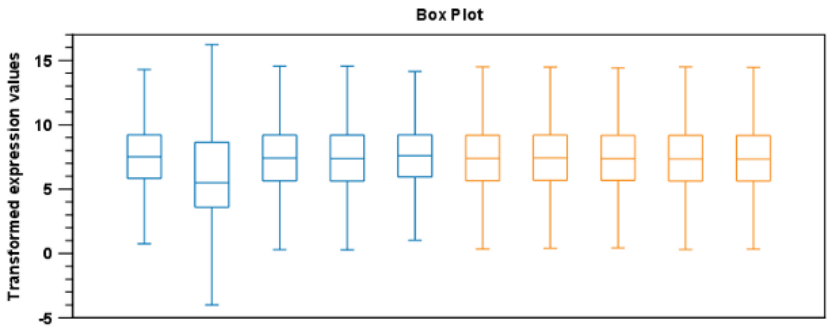


Figure 30.58: *Box plot for a two-group experiment with 5 samples.*

2. calculating pairwise distances between all clusters
3. joining the two closest clusters into one new cluster
4. iterating 2-3 until there is only one cluster left (which will contain all samples).

The tree is drawn so that the distances between clusters are reflected by the lengths of the branches in the tree. Thus, features with expression profiles that closely resemble each other have short distances between them, those that are more different, are placed further apart.

(See [Eisen et al., 1998] for a classical example of application of a hierarchical clustering algorithm in microarray analysis. The example is on features rather than samples).

To start the clustering:

> **Toolbox** | **Microarray and Small RNA 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 30.59. 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.

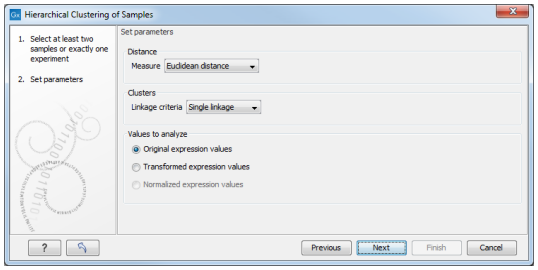


Figure 30.59: *Parameters for hierarchical clustering of samples.*

At the top, you can choose three kinds of **Distance measures**:

- **Euclidean distance**. The ordinary distance between two points - the length of the segment connecting them. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean

distance between u and v is

$$|u-v| = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}.$$

- **1 - Pearson correlation**. The Pearson correlation coefficient between two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ is defined as

$$r = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x_i - \overline{x}}{s_x}) * (\frac{y_i - \overline{y}}{s_y})$$

  where $\overline{x}/\overline{y}$ is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance**. The Manhattan distance between two points is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between u and v is

$$|u-v| = \sum_{i=1}^{n}|u_i - v_i|.$$

Next, you can select the cluster linkage to be used:

- **Single linkage**. The distance between two clusters is computed as the distance between the two closest elements in the two clusters.
- **Average linkage**. The distance between two clusters is computed as the average distance between objects from the first cluster and objects from the second cluster. The averaging is performed over all pairs (x, y), where x is an object from the first cluster and y is an object from the second cluster.
- **Complete linkage**. The distance between two clusters is computed as the maximal object-to-object distance $d(x_i, y_j)$, where x_i comes from the first cluster, and y_j comes from the second cluster. In other words, the distance between two clusters is computed as the distance between the two farthest objects in the two clusters.

At the bottom, you can select which values to cluster (see section 30.4.1).

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.

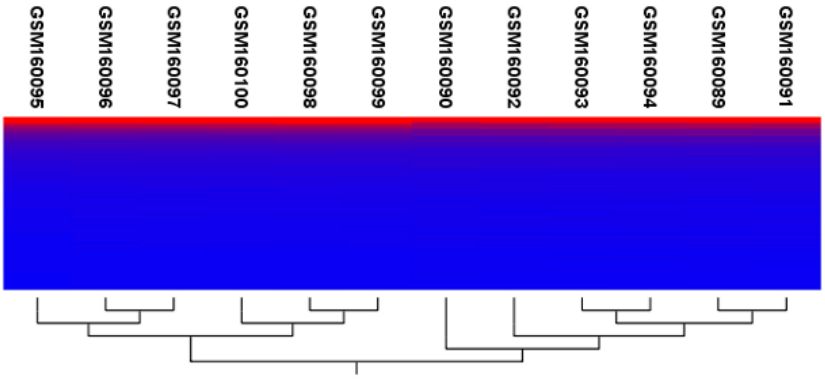


Figure 30.60: *Sample clustering.*

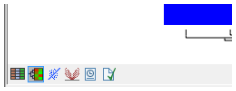


Figure 30.61: *Showing the hierarchical clustering of an experiment.*

Result of hierarchical clustering of samples

The result of a sample clustering is shown in figure 30.60.

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 30.61).

If you have run the workflow version of the tool, or selected a number of **samples** (( ) or ( )) as input, a new element will be created that has to be saved separately.

Regardless of the input, the view of the clustering is the same. As you can see in figure 30.60, 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 30.7.1).

Researchers often have a priori knowledge of which samples in a study should be similar (e.g. samples from the same experimental condition) and which should be different (samples from biological distinct conditions). Thus, researches have expectations about how they should cluster. Samples that are placed unexpectedly in the hierarchical clustering tree may be samples that have been wrongly allocated to a group, samples of unintended or unclean tissue composition or samples for which the processing has gone wrong. Unexpectedly placed samples, of course, could also be highly interesting samples.

There are a number of options to change the appearance of the heat map. At the top of the **Side Panel**, you find the **Heat map** preference group (see figure 30.62).

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 30.75).

Note that if you perform an identical clustering, the existing heat map will simply be replaced.

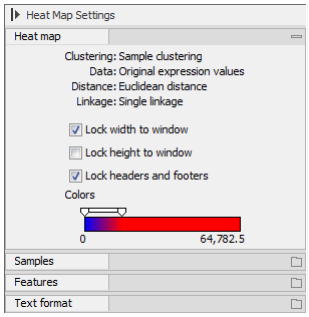


Figure 30.62: *Side Panel of heat map.*

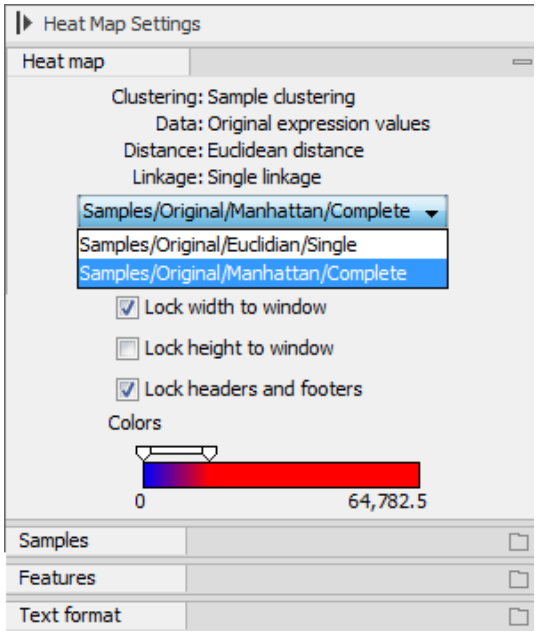


Figure 30.63: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

Below this box, there is a number of settings for displaying the heat map.

- **Lock width to window**. When you zoom in the heat map, you will per default only zoom in on the vertical level. This is because the width of the heat map is locked to the window. If you uncheck this option, you will zoom both vertically and horizontally. Since you always have more features than samples, it is useful to lock the width since you then have all the samples in view all the time.
- **Lock height to window**. This is the corresponding option for the height. Note that if you check both options, you will not be able to zoom at all, since both the width and the height is fixed.
- **Lock headers and footers**. This will ensure that you are always able to see the sample and feature names and the trees when you zoom in.
- **Colors**. The expression levels are visualized using a gradient color scheme, where the right side color is used for high expression levels and the left side color is used for low expression levels. You can change the coloring by clicking the box, and you can change the relative coloring of the values by dragging the two knobs on the white slider above.

Below you find the **Samples** and **Features** groups. They contain options to show names, legend, and tree above or below the heatmap. Note that for clustering of samples, you find the tree options in the **Samples** group, and for clustering of features, you find the tree options in the **Features** group. With the tree options, you can also control the **Tree size**, from tiny to very large, and the option of showing the full tree, no matter how much space it will use.

Note that if you wish to use the same settings next time you open a heat map, you need to save the settings of the **Side Panel** (see section 4.6).

30.5.3 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** | **Microarray and Small RNA 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 30.64.

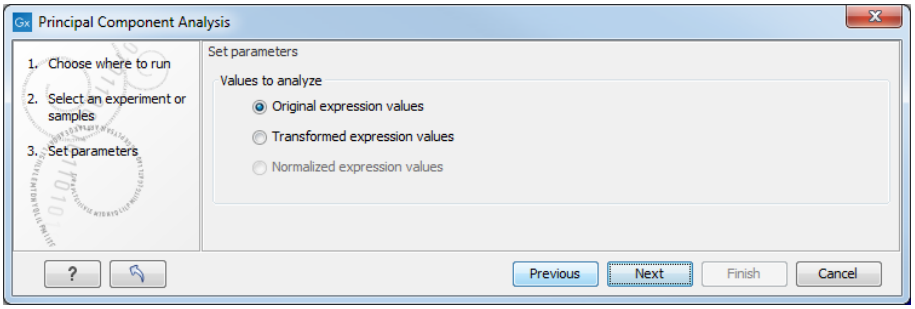


Figure 30.64: *Selecting which values the principal component analysis should be based on.*

In this dialog, you select the values to be used for the principal component analysis (see section 30.4.1).

Click **Finish** to start the tool.

Principal component analysis plot

This will create a principal component plot as shown in figure 30.65.

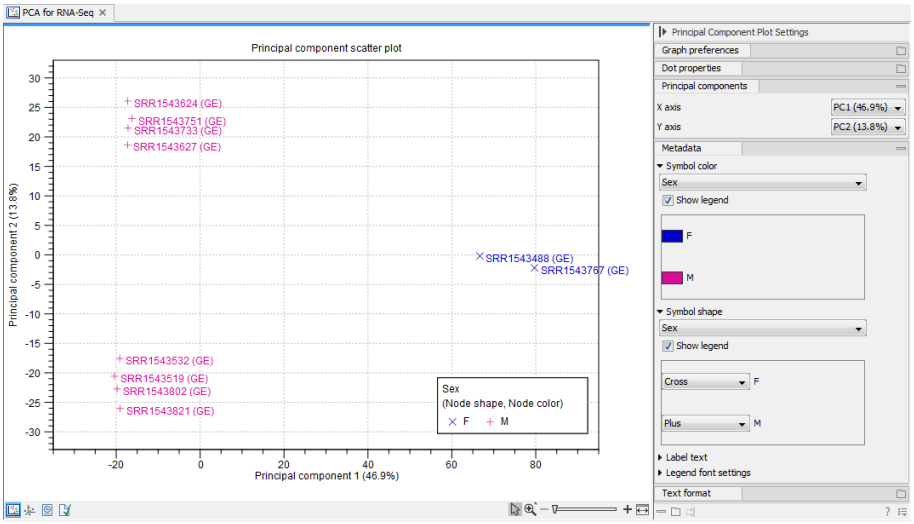


Figure 30.65: *A principal component analysis colored by group.*

The plot shows the projection of the samples onto the two-dimensional space spanned by the first and second principal component of the covariance matrix. In the bottom part of the side-panel, the 'Projection/Correlation' part, you can change to show the projection onto the *correlation* matrix rather than the *covariance* matrix by choosing 'Correlation scatter plot'. Both plots will show how the samples separate along the two directions between which the samples exhibit the largest amount of variation. For the 'projection scatter plot' this variation is measured in absolute terms, and depends on the units in which you have measured your samples. The correlation scatter plot is a normalized version of the projection scatter plot, which makes it possible to compare principal component analysis between experiments, even when these have not been done using the same units (e.g an experiment that uses 'original' scale data and another one that uses 'log-scale' data).

The plot in figure 30.65 is based on a two-group experiment. The group relationships are indicated by color. We expect the samples from within a group to exhibit less variability when compared, than samples from different groups. Thus samples should cluster according to groups and this is what we see. The PCA plot is thus helpful in identifying outlying samples and samples that have been wrongly assigned to a group.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside

  - Inside

- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks

- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

- **y = 0 axis**. Draws a line where y = 0. Below there are some options to control the appearance of the line:
  - **Line width**
    * Thin
    * Medium
    * Wide
  - **Line type**
    * None
    * Line
    * Long dash
    * Short dash
  - **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties**:

- **Drop down menu** In this you choose which of the samples (that is, which 'dots') the choices you make below should apply to. You can choose between 'All', a particular group in your experiment, or a particular samples in your experiment.

- **Select sample or group**. When you wish to adjust the properties below, first select an item in this drop-down menu. That will apply the changes below to this item. If your plot is based on an experiment, the drop-down menu includes both group names and sample names, as well as an entry for selecting "All". If your plot is based on single elements, only sample names will be visible. Note that there are sometimes "mixed states" when you select a group where two of the samples e.g. have different colors. Selecting a new color in this case will erase the differences.

- **Dot type**
  - None
  - Cross
  - Plus

  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot

- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Show name**. This will show a label with the name of the sample next to the dot. Note that the labels quickly get crowded, so that is why the names are not put on per default.

Note that if you wish to use the same settings next time you open a principal component plot, you need to save the settings of the **Side Panel** (see section 4.6).

Scree plot

Besides the view shown in figure 30.65, the result of the principal component can also be viewed as a scree plot by clicking the **Show Scree Plot** ( ) button at the bottom of the view. The scree plot shows the proportion of variation in the data explained by each of the principal components. The first principal component accounts for the largest part of the variability.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

The **Lines and plots** below contains the following parameters:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Line width**
  - Thin
  - Medium
  - Wide
- **Line type**
  - None
  - Line
  - Long dash
  - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that the graph title and the axes titles can be edited simply by clicking them with the mouse. These changes will be saved when you **Save** () the graph - whereas the changes in the **Side Panel** need to be saved explicitly (see section 4.6).

30.6 Statistical analysis - identifying differential expression

The *Biomedical Genomics Workbench* is designed to help you identify differential expression.

30.6.1 Empirical analysis of DGE

The Empirical analysis of DGE tool implements the 'Exact Test' for two-group comparisons developed by Robinson and Smyth [Robinson and Smyth, 2008] and incorporated in the EdgeR Bioconductor package [Robinson et al., 2010]. The test is applicable to count data only, and is designed specifically to deal with situations in which *many* features are studied simultaneously (e.g. genes in a genome) but where only a *few* biological replicates are available for each of the experimental groups studied. This is typically the case for RNA-seq expression analysis.

The test uses the raw counts, and implicitly carries out normalization and transformation of these counts (see below for details). It is based on the assumption that the count data follows a Negative Binomial distribution, which in contrast to the Poisson distribution has the characteristic that it allows for a non-constant mean-variance relationship. The test is also appropriate for larger numbers of samples.

The 'Exact Test' of Robinson and Smyth is similar to Fisher's Exact Test, but also accounts for overdispersion caused by biological variability. Whereas Fisher's Exact Test compares the counts in one sample against those of another, the 'Exact Test' compares the counts in one *set* of count samples against those in another *set* of count samples. This is achieved by replacing the Hypergeometric distributions of Fisher's Exact Test by Negative binomial distributions, whereby the variability within each of the two groups of samples compared is taken into account. This only works if the dispersions in the two groups compared are identical. As this cannot generally be assumed to be the case for the *original* (nor for the normalized) data, pseudodata for which the dispersion *is* identical is generated from the original data, and the test is carried out on this pseudodata. The generation of the pseudodata is performed simultaneously with the estimation of the dispersion, in an iterative procedure called quantile-adjusted conditional maximum likelihood. Either a single common dispersion for all features may be assumed (as in [Robinson and Smyth, 2008]), or it may be assumed that the dispersion for each feature (e.g. gene) is a 'weighted average' of the common dispersion and feature (e.g. gene) specific dispersions (as suggested in [Robinson and Smyth, 2007]). The weight given to each of the components depends on the number of samples in the groups: the more samples there are in the groups, the higher the weight will be given to the gene-specific component.

The Exact Test in the EdgeR Bioconductor package provides the user with the option to set a large number of parameters. The implementation of the 'Empirical analysis of DGE' algorithm in the Genomics Workbench uses for the most parts the default settings in the edgeR package, version 3.4.0. A detailed outline of the parameter settings is given in section 30.6.1).

Empirical analysis of DGE - implementation parameters

The 'Empirical analysis of DGE' algorithm in the *Biomedical Genomics Workbench* is a re-implementation of the "Exact Test", available as part of the EdgeR Bioconductor package.

The parameter values used in the *Biomedical Genomics Workbench* implementation are the default values for the equivalent parameters in the EdgeR Bioconductor implementation in all but one case. The exception is the estimateCommonDisp tol parameter, where the default is more stringent than that of EdgeR. The advantage of using a more stringent value for this parameter is that the results will be more accurate. The disadvantage is that the algorithm will be slightly slower, however according to our performance tests, this change has only a marginal impact on the run time of the tool.

The parameter values used in the *Biomedical Genomics Workbench* implementation, with reference to the EdgeR function names for clarity, are provided in the table below.

Function in BioC package	Parameter name	Value used and comments
calcNormFactors	method	"TMM"
	refColumn	NULL (automatically selected)
	logratioTrim	0.3
	sumTrim	0.05
	doWeighting	TRUE
	Acutoff	-1e10
estimateCommonDisp	tol	1e-14 (default in edgeR: 1e-6)
	rowsum.filter	Set by user in wizard ("Total count filter cutoff", default 5)
estimateTagewiseDisp	prior.df	10
	trend	"movingave"
	span	NULL
	method	"grid"
	grid.length	11
	grid.range	c(-6, 6)
mglmOneGroup	maxit	50
	tol	1e-10
aveLogCPM	prior.count	2
	dispersion	0.05
exactTest	pair	Set by user in wizard ("Exact test comparisons")
	dispersion	"auto" (tagwise if available, otherwise common)
	rejection.region	"doubletail"
	big.count	900
	prior.count	0.125

Running the Empirical analysis of DGE

First, find the **Empirical analysis of DGE** tool:

> **Toolbox** | **Microarray and Small RNA Analysis (**  **)**| **Statistical Analysis** | **Empirical Analysis of DGE** ()

The original count data for a full expression experiment are the expected input to the Empirical Analysis of DGE tool.

When Experiments created within the Workbench are used as input, the original count values are always used. Columns of such Experiments that contain transformed or normalized values are ignored.

If expression values are being imported from outside the Workbench for use with this test, the data should be original (non-transformed, non-normalized) counts.

Whether the data has been generated in the Workbench or outside the Workbench and imported, the full set of expression results should be used. Please do not run this test on a subset of values from the original sample data.

The reason that the complete set of original count data for samples should be used as input to this test is that the algorithm assumes that the counts on which it operates are Negative

Binomially distributed. It implicitly normalizes and transforms these counts, so if the counts have been altered prior to submitting them to the Empirical Analysis of DGE tool, this assumption is likely to be compromised.

When running the Empirical analysis of DGE tool in the Genomics workbench, the user is asked to specify two parameters related to the estimation of the dispersion (figure 30.66). Of these, the 'Total count filter cut-off' specifies which features should be considered when estimating the common dispersion component. Features for which the counts across all samples are low are likely to contribute mostly with noise to the estimation, and features with a lower cummulative count across samples than the value specified will be ignored. When the check-box 'Estimate tag-wise dispersions' is checked, the dispersion estimate for each gene will be a weighted combination of the tag-wise and common dispersion, if the check-box is un-ticked the common dispersion will be used for all genes.

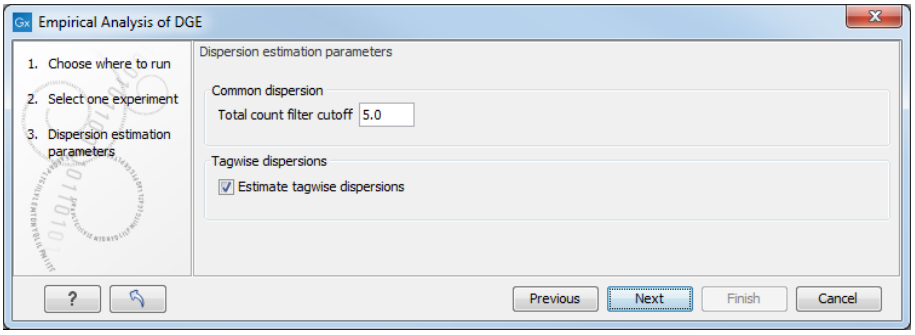


Figure 30.66: *Empirical analysis of DGE: setting the parameters related to dispersion.*

The Empirical analysis of DGE may be carried out between all pairs of groups (by clicking the 'All pairs' button) or for each group against a specified reference group (by clicking the 'Against reference' button) (figure 30.67). 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). Foe example, the All pairs option should be selected when you wish to perform the test of equality for group means for all of the pairs, e.g. if you would like to compare different tissues where each tissue is represented in a group. In this case there is no reference group, so the following comparisons will be performed:

- liver vs heart
- liver vs lung
- heart vs lung

The Against reference option should be selected when you wish to perform the test of equality for group means against one group, the reference, rather than all groups as above. Against reference could be used if you have a wild type and some mutant groups, e.g. Wild type, Mutant 1 and Mutant 2. In this case you might be interested in comparing the mutants to the wild type, but comparing the mutants to each other is not of interest. In this case the Wild Type group is considered the reference and the comparisons will be perfomed:

- Wild type vs Mutant 1
- Wild type vs Mutant 2

Note that with the Against reference option fewer comparisons are made, as in the above example where Mutant 1 vs. Mutant 2 is not considered.

Below you can select to add two kinds of corrected p-values to the analysis (in addition to the standard p-value produced for the test statistic):

- **Bonferroni corrected**.
- **FDR corrected**.

Both are calculated from the original p-values, and aim in different ways to take into account the issue of multiple testing [Dudoit et al., 2003]. The problem of multiple testing arises because the original p-values are related to a single test: the p-value is the probability of observing a more extreme value than that observed in the test carried out. If the p-value is 0.04, we would expect an as extreme value as that observed in 4 out of 100 tests carried out among groups with no difference in means. Popularly speaking, if we carry out 10000 tests and select the features with original p-values below 0.05, we will expect about 0.05 times 10000 = 500 to be false positives.

The Bonferroni corrected p-values handle the multiple testing problem by controlling the 'family-wise error rate': the probability of making at least one false positive call. They are calculated by multiplying the original p-values by the number of tests performed. The probability of having at least one false positive among the set of features with Bonferroni corrected p-values below 0.05, is less than 5%. The Bonferroni correction is conservative: there may be many genes that are differentially expressed among the genes with Bonferroni corrected p-values above 0.05, that will be missed if this correction is applied.

Instead of controlling the family-wise error rate we can control the false discovery rate: FDR. The false discovery rate is the proportion of false positives among all those declared positive. We expect 5 % of the features with FDR corrected p-values below 0.05 to be false positive. There are many methods for controlling the FDR - the method used in *Biomedical Genomics Workbench* is that of [Benjamini and Hochberg, 1995].

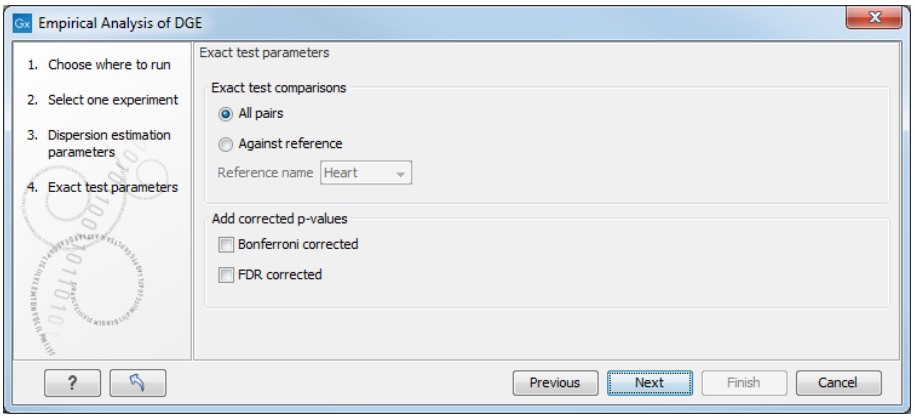


Figure 30.67: *Empirical analysis of DGE: setting comparisons and corrected p-value options.*

When the Empirical analysis of DGE is run three columns will be added to the experiment table for each pair of groups that are analyzed: the 'P-value', 'Fold change' and 'Weighted difference' columns. The 'P-value' holds the p-value for the Exact test. The 'Fold Change' and 'Weighted difference' columns are both calculated from the estimated relative abundances, which are derived internally in the Exact Test algorithm. They depend on both the sizes (depth of

coverage/library size) of the samples, the magnitude of the counts and on the estimated negative binomial dispersion, so they cannot be obtained from the original counts by simple algebraic calculations.

The 'Fold Change' will tell you how many times bigger the relative abundance of group 2 is relative to that of group 1. If the relative abundance of group 2 is bigger than that of group 1 the fold change is the relative abundance of group 2 divided by that of group 1. If the relative abundance of group 2 is smaller than that of group 1 the fold change is the relative abundance of group 1 divided by that of group 2 with a negative sign. The 'weighted difference' column contains the difference between the relative abundance of group 2 and the relative abundance of group 1. In addition to the three automatically added columns, columns containing the Bonferroni and FDR corrected p-values will be added if that was specified by the user.

30.6.2 Tests on proportions

The proportions-based tests are applicable in situations where your data samples consists of counts of a number of 'types' of data. This could e.g. be in a study where gene expression levels are measured by RNA-Seq or tag profiling. Here the different 'types' could correspond to the different 'genes' in a reference genome, and the counts could be the numbers of reads matching each of these genes. The tests compare counts by considering the proportions that they make up the total sum of counts in each sample. By comparing the expression levels at the level of proportions rather than raw counts, the data is corrected for sample size.

There are two tests available for comparing proportions: the test of [Kal et al., 1999] and the test of [Baggerly et al., 2003]. Both tests compare pairs of groups. If you have a multi-group experiment (see section 30.2.1), 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 (see 30.6.4).

Baggerley et al.'s test (Beta-binomial)

Baggerley et al.'s test [Baggerly et al., 2003] compares the proportions of counts in a group of samples against those of another group of samples, and is suited to cases where replicates are available in the groups. The samples are given different weights depending on their sizes (total counts). The weights are obtained by assuming a Beta distribution on the proportions in a group, and estimating these, along with the proportion of a binomial distribution, by the method of moments. The result is a weighted t-type test statistic.

When Baggerley's test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Weighted proportions difference' column contains the difference between the mean of the weighted proportions across the samples assigned to group 2 and the mean of the weighted proportions across the samples assigned to group 1. The 'Weighted proportions fold change' column tells you how many times bigger the mean of the weighted proportions in group 2 is relative to that of group 1. If the mean of the weighted proportions in group 2 is bigger than that in group 1 this value is the mean of the weighted proportions in group 2 divided by that in group 1. If the mean of the weighted proportions in group 2 is smaller than that in group 1 the fold change is the mean of the weighted proportions in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 30.6.4).

30.6.3 Gaussian-based tests

The tests based on the Gaussian distribution essentially compare the mean expression level in the experimental groups in the study, and evaluates the significance of the difference relative to the variance (or 'spread') of the data within the groups. The details of the formula used for calculating the test statistics vary according to the experimental setup and the assumptions you make about the data (read more about this in the sections on t-test and ANOVA below). The explanation of how to proceed is divided into two, depending on how many groups there are in your experiment. First comes the explanation for t-tests which is the only analysis available for two-group experimental setups (t-tests can also be used for pairwise comparison of groups in multi-group experiments). Next comes an explanation of the ANOVA test which can be used for multi-group experiments.

Note that the test statistics for the t-test and ANOVA analysis use the estimated group variances in their denominators. If all expression values in a group are identical the estimated variance for that group will be zero. If the estimated variances for both (or all) groups are zero the denominator of the test statistic will be zero. The numerator's value depends on the difference of the group means. If this is zero, the numerator is zero and the test statistic will be 0/0 which is NaN. If the numerator is different from zero the test statistic will be + or - infinity, depending on which group mean is bigger. If all values in all groups are identical the test statistic is set to zero.

T-tests

For experiments with two groups you can, among the Gaussian tests, only choose a **T-test** as shown in figure 30.68.

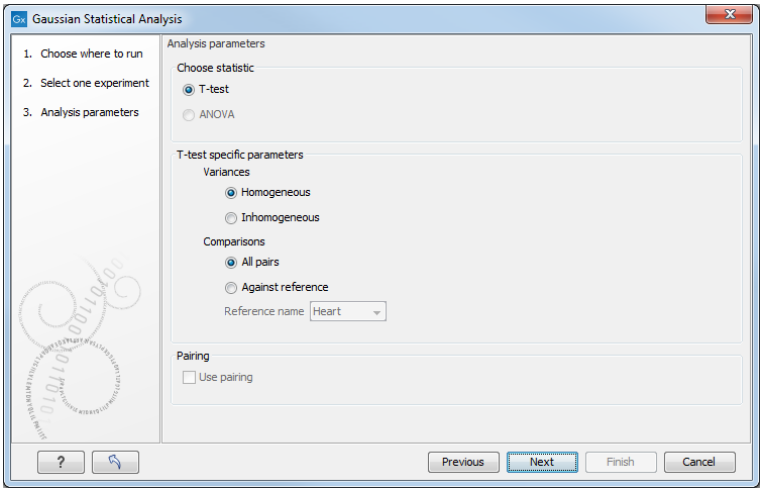


Figure 30.68: *Selecting a t-test.*

There are different types of t-tests, depending on the assumption you make about the variances in the groups. By selecting 'Homogeneous' (the default) calculations are done assuming that the groups have equal variances. When 'In-homogeneous' is selected, this assumption is not made.

The t-test can also be chosen if you have a multi-group experiment. In this case you may choose either to have t-tests produced for all pairs of groups (by clicking the 'All pairs' button) or to have a t-test produced for each group compared to a specified reference group (by clicking the 'Against reference' button). In the last case you must specify which of the groups you want to use as reference (the default is to use the group you specified as Group 1 when you set up the experiment).

If a experiment with pairing was set up (see section 30.2.1) the **Use pairing** tick box is active. If ticked, paired t-tests will be calculated, if not, the formula for the standard t-test will be used.

When a t-test is run on an experiment four columns will be added to the experiment table for each pair of groups that are analyzed. The 'Difference' column contains the difference between the mean of the expression values across the samples assigned to group 2 and the mean of the expression values across the samples assigned to group 1. The 'Fold Change' column tells you how many times bigger the mean expression value in group 2 is relative to that of group 1. If the mean expression value in group 2 is bigger than that in group 1 this value is the mean expression value in group 2 divided by that in group 1. If the mean expression value in group 2 is smaller than that in group 1 the fold change is the mean expression value in group 1 divided by that in group 2 with a negative sign. The 'Test statistic' column holds that value of the test statistic, and the 'P-value' holds the two-sided p-value for the test. Up to two more columns may be added if the options to calculate Bonferroni and FDR corrected p-values were chosen (see 30.6.4).

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 30.2.1) 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 (see 30.6.4).

30.6.4 Corrected p-values

Clicking **Next** will display a dialog as shown in figure 30.69.

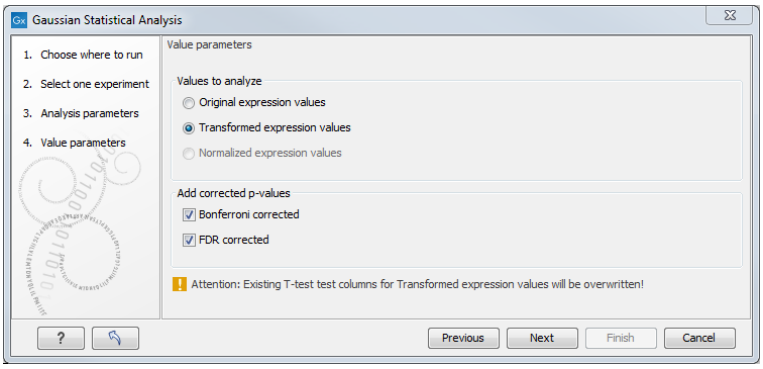


Figure 30.69: *Additional settings for the statistical analysis.*

At the top, you can select which values to analyze (see section 30.4.1).

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 *Biomedical Genomics 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.

30.6.5 Volcano plots - inspecting the result of the statistical analysis

The results of the statistical analysis are added to the experiment and can be shown in the experiment table (see section 30.2.2). Typically columns containing the differences (or weighted differences) of the mean group values and the fold changes (or weighted fold changes) of the mean group values will be added along with a column of p-values. Also, columns with FDR or Bonferroni corrected p-values will be added if these were calculated. This added information allows features to be sorted and filtered to exclude the ones without sufficient proof of differential expression (learn more in section 3.3).

If you want a more visual approach to the results of the statistical analysis, you can click the **Show Volcano Plot** () button at the bottom of the experiment table view. In the same way as the scatter plot presented in section 30.2.4, 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 30.70.

The volcano plot shows the relationship between the p-values of a statistical test and the magnitude of the difference in expression values of the samples in the groups. On the y-axis the $-\log_{10}$ p-values are plotted. For the x-axis you may choose between two sets of values by choosing either 'Fold change' or 'Difference' in the volcano plot side panel's 'Values' part. If you choose 'Fold change' the log of the values in the 'fold change' (or 'Weighted fold change') column for the test will be displayed. If you choose 'Difference' the values in the 'Difference' (or 'Weighted difference') column will be used. Which values you wish to display will depend upon the scale of you data (Read the note on fold change in section 30.2.2).

The larger the difference in expression of a feature, the more extreme it's point will lie on

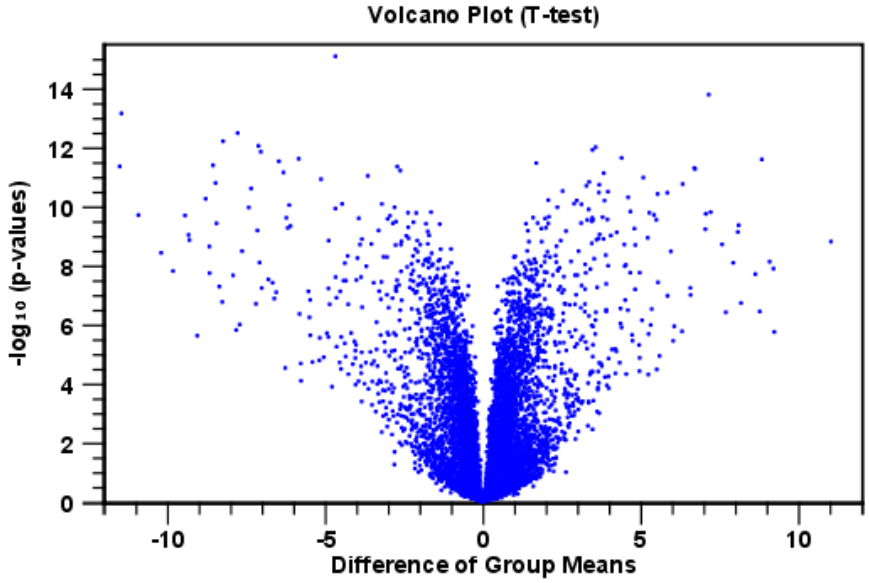


Figure 30.70: *Volcano plot.*

the X-axis. The more significant the difference, the smaller the p-value and thus the higher the $-\log_{10}(p)$ value. Thus, points for features with highly significant differences will lie high in the plot. Features of interest are typically those which change significantly and by a certain magnitude. These are the points in the upper left and upper right hand parts of the volcano plot.

If you have performed different tests or you have an experiment with multiple groups you need to specify for which test and which group comparison you want the volcano plot to be shown. You do this in the 'Test' and 'Values' parts of the volcano plot side panel.

Options for the volcano plot are described in further detail when describing the **Side Panel** below.

If you place your mouse on one of the dots, a small text box will tell the name of the feature. Note that you can zoom in and out on the plot (see section 2.2).

In the **Side Panel** to the right, there is a number of options to adjust the view of the volcano plot. Under **Graph preferences**, you can adjust the general properties of the volcano plot

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.
- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None

  - Major ticks

- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.

Below the general preferences, you find the **Dot properties**, where you can adjust coloring and appearance of the dots.

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

At the very bottom, you find two groups for choosing which values to display:

- **Test**. In this group, you can select which kind of test you want the volcano plot to be shown for.
- **Values**. Under **Values**, you can select which values to plot. If you have multi-group experiments, you can select which groups to compare. You can also select whether to plot **Difference** or **Fold change** on the x-axis. Read the note on fold change in section 30.2.2.

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).

30.7 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.

30.7.1 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** | **Microarray and Small RNA 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 30.2.2.

Clicking **Next** will display a dialog as shown in figure 30.71. The hierarchical clustering algorithm requires that you specify a distance measure and a cluster linkage. The distance measure is used specify how distances between two features should be calculated. The cluster linkage specifies how you want the distance between two clusters, each consisting of a number of features, to be calculated.

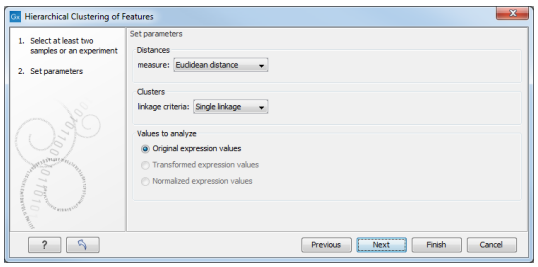


Figure 30.71: *Parameters for hierarchical clustering of features.*

At the top, you can choose three kinds of **Distance measures**:

- **Euclidean distance**. The ordinary distance between two points - the length of the segment connecting them. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Euclidean

distance between u and v is

$$|u - v| = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}.$$

- **1 - Pearson correlation**. The Pearson correlation coefficient between two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ is defined as

$$r = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{x_i - \overline{x}}{s_x}) * (\frac{y_i - \overline{y}}{s_y})$$

  where $\overline{x}/\overline{y}$ is the average of values in x/y and s_x/s_y is the sample standard deviation of these values. It takes a value $\in [-1, 1]$. Highly correlated elements have a high absolute value of the Pearson correlation, and elements whose values are un-informative about each other have Pearson correlation 0. Using $1 - |Pearsoncorrelation|$ as distance measure means that elements that are highly correlated will have a short distance between them, and elements that have low correlation will be more distant from each other.

- **Manhattan distance**. The Manhattan distance between two points is the distance measured along axes at right angles. If $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$, then the Manhattan distance between u and v is

$$|u - v| = \sum_{i=1}^{n}|u_i - v_i|.$$

Next, you can select different ways to calculate distances between clusters. The possible cluster linkage to use are:

- **Single linkage**. The distance between two clusters is computed as the distance between the two closest elements in the two clusters.
- **Average linkage**. The distance between two clusters is computed as the average distance between objects from the first cluster and objects from the second cluster. The averaging is performed over all pairs (x, y), where x is an object from the first cluster and y is an object from the second cluster.
- **Complete linkage**. The distance between two clusters is computed as the maximal object-to-object distance $d(x_i, y_j)$, where x_i comes from the first cluster, and y_j comes from the second cluster. In other words, the distance between two clusters is computed as the distance between the two farthest objects in the two clusters.

At the bottom, you can select which values to cluster (see section 30.4.1). Click **Finish** to start the tool.

Result of hierarchical clustering of features

The result of a feature clustering is shown in figure 30.72.

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 30.73).

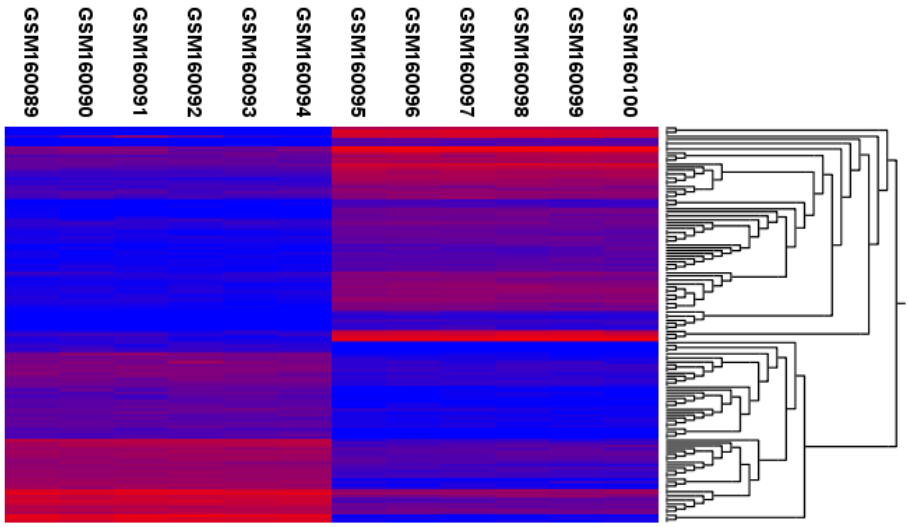


Figure 30.72: *Hierarchical clustering of features.*

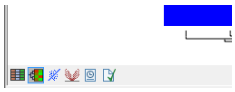


Figure 30.73: *Showing the hierarchical clustering of an experiment.*

If you have selected a number of **samples** (( ) or ( )) as input, a new element will be created that has to be saved separately.

Regardless of the input, a hierarchical tree view with associated heatmap is produced (figure 30.72). 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 30.74).

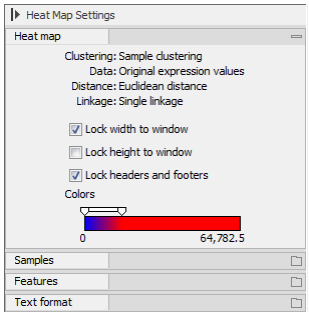


Figure 30.74: *Side Panel of heat map.*

At the top, there is information about the heat map currently displayed. The information regards type of clustering, expression value used together with distance and linkage information. If you

have performed more than one clustering, you can choose between the resulting heat maps in a drop-down box (see figure 30.75).

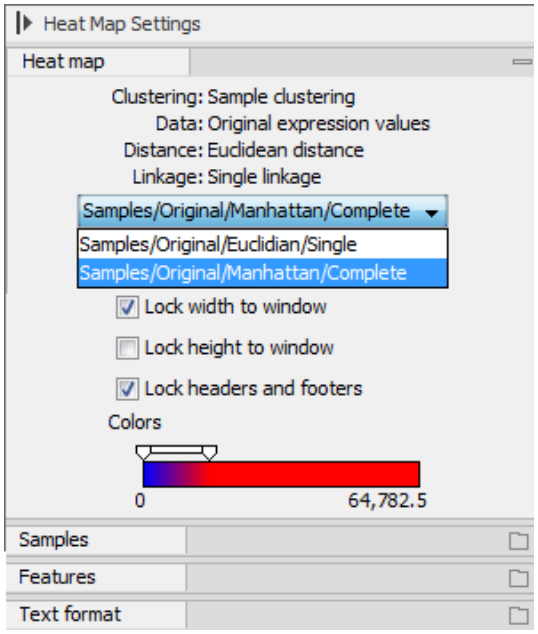


Figure 30.75: *When more than one clustering has been performed, there will be a list of heat maps to choose from.*

Note that if you perform an identical clustering, the existing heat map will simply be replaced. Below this box, there is a number of settings for displaying the heat map.

- **Lock width to window**. When you zoom in the heat map, you will per default only zoom in on the vertical level. This is because the width of the heat map is locked to the window. If you uncheck this option, you will zoom both vertically and horizontally. Since you always have more features than samples, it is useful to lock the width since you then have all the samples in view all the time.
- **Lock height to window**. This is the corresponding option for the height. Note that if you check both options, you will not be able to zoom at all, since both the width and the height is fixed.
- **Lock headers and footers**. This will ensure that you are always able to see the sample and feature names and the trees when you zoom in.
- **Colors**. The expression levels are visualized using a gradient color scheme, where the right side color is used for high expression levels and the left side color is used for low expression levels. You can change the coloring by clicking the box, and you can change the relative coloring of the values by dragging the two knobs on the white slider above.

Below you find the **Samples** and **Features** groups. They contain options to show names, legend, and tree above or below the heatmap. Note that for clustering of samples, you find the tree options in the **Samples** group, and for clustering of features, you find the tree options in the **Features** group. With the tree options, you can also control the **Tree size**, from tiny to very large, and the option of showing the full tree, no matter how much space it will use.

Note that if you wish to use the same settings next time you open a heat map, you need to save the settings of the **Side Panel** (see section 4.6).

30.7.2 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 | Microarray and Small RNA Analysis ( )| Feature Clustering | K-means/medoids Clustering ( )

Select at least two samples (( ) or ( )) or an experiment ( ).

Note! If your data contains many features, the clustering will take very long time and could make your computer unresponsive. It is recommended to perform this analysis on a subset of the data (which also makes it easier to make sense of the clustering). See how to create a sub-experiment in section 30.2.2.

Clicking **Next** will display a dialog as shown in figure 30.76.

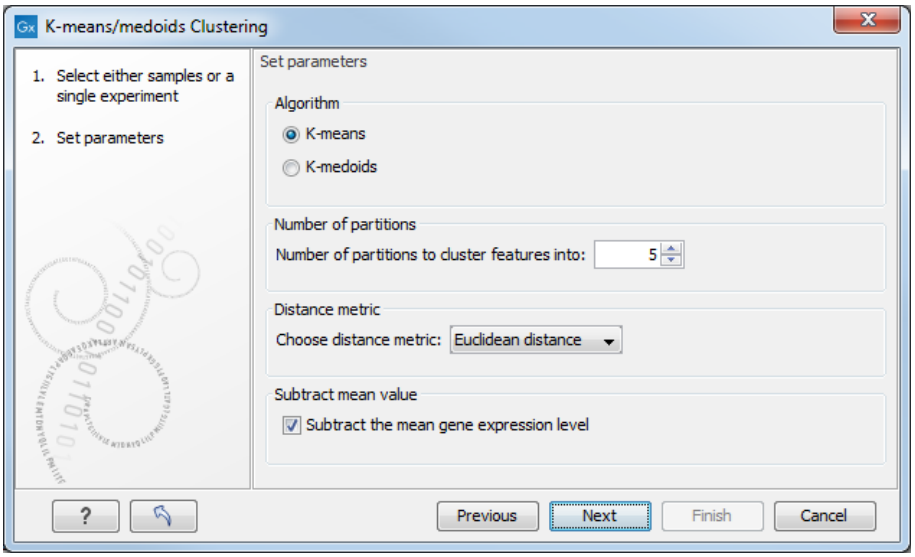


Figure 30.76: *Parameters for k-means/medoids clustering.*

The parameters are:

- **Algorithm**. You can choose between two clustering methods:
  - **K-means**. K-means clustering assigns each point to the cluster whose center is nearest. The center/centroid of a cluster is defined as the average of all points in the cluster. If a data set has three dimensions and the cluster has two points $X = (x_1, x_2, x_3)$ and $Y = (y_1, y_2, y_3)$, then the centroid Z becomes $Z = (z_1, z_2, z_3)$, where $z_i = (x_i + y_i)/2$ for $i = 1, 2, 3$. The algorithm attempts to minimize the intra-cluster variance defined by:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2$$

where there are k clusters $S_i, i = 1, 2, \ldots, k$ and μ_i is the centroid of all points $x_j \in S_i$. The detailed algorithm can be found in [Lloyd, 1982].

- **K-medoids**. K-medoids clustering is computed using the PAM-algorithm (PAM is short for Partitioning Around Medoids). It chooses datapoints as centers in contrast to the K-means algorithm. The PAM-algorithm is based on the search for k representatives (called medoids) among all elements of the dataset. When having found k representatives k clusters are now generated by assigning each element to its nearest medoid. The algorithm first looks for a good initial set of medoids (the BUILD phase). Then it finds a local minimum for the objective function:

$$V = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - c_i)^2$$

where there are k clusters $S_i, i = 1, 2, \ldots, k$ and c_i is the medoid of S_i. This solution implies that there is no single switch of an object with a medoid that will decrease the objective (this is called the SWAP phase). The PAM-agorithm is described in [Kaufman and Rousseeuw, 1990].

- **Number of partitions**. The 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 30.77.

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 30.4.1).

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

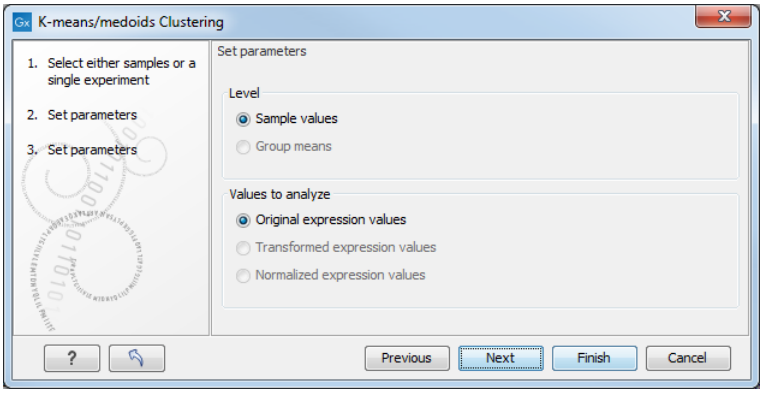


Figure 30.77: *Parameters for k-means/medoids clustering.*

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 30.76) - there is one graph per cluster. Using drag and drop as explained in section 2.1.6, you can arrange the views to see more than one graph at the time.

Figure 30.78 shows an example where four clusters have been arranged side-by-side.

The samples used are from a time-series experiment, and you can see that the expression levels for each cluster have a distinct pattern. The two clusters at the bottom have falling and rising expression levels, respectively, and the two clusters at the top both fall at the beginning but then rise again (the one to the right starts to rise earlier that the other one).

Having inspected the graphs, you may wish to take a closer look at the features represented in each cluster. In the experiment table, the clustering has added an extra column with the name of the cluster that the feature belongs to. In this way you can filter the table to see only features from a specific cluster. This also means that you can select the feature of this cluster in a volcano or scatter plot as described in section 30.2.5.

30.8 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 30.2.3.

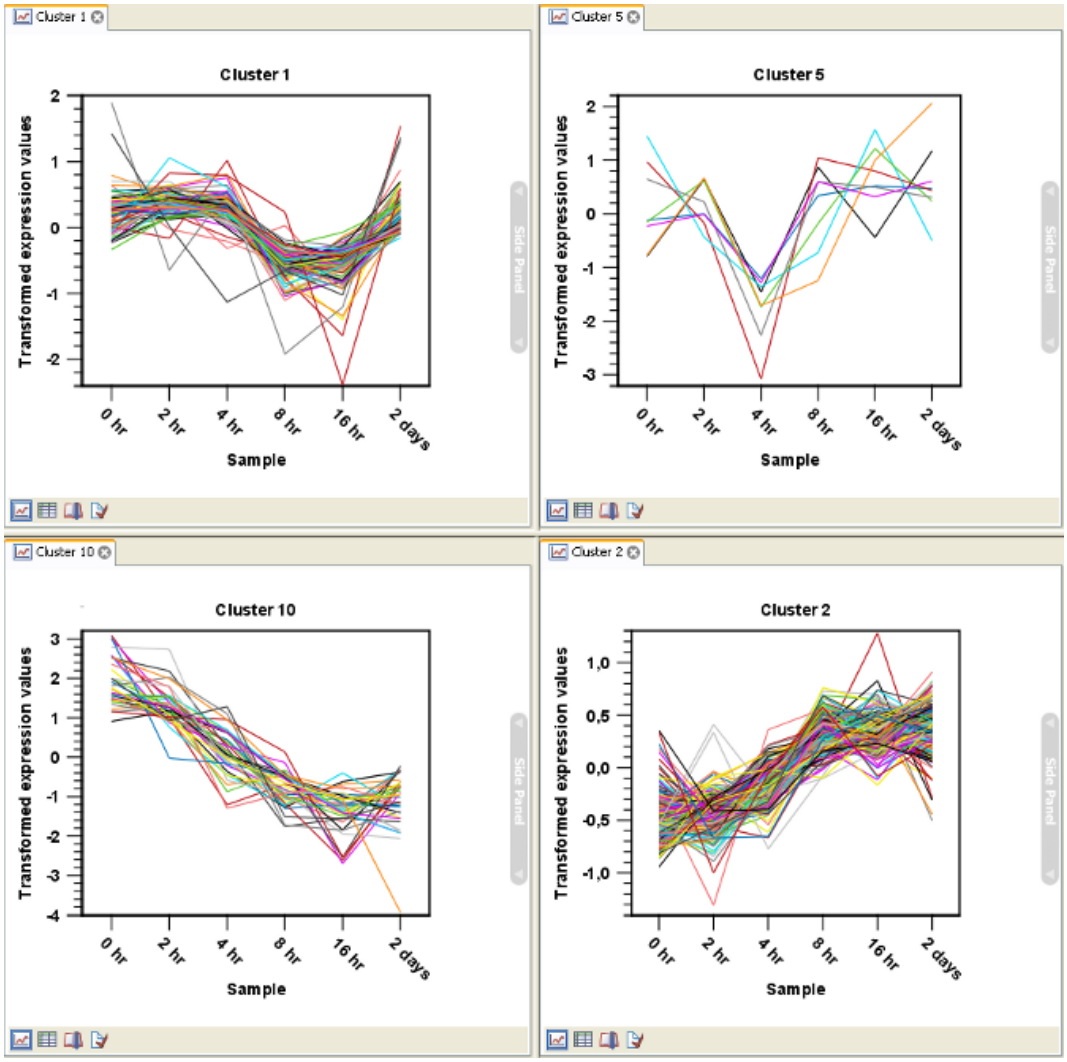


Figure 30.78: *Four clusters created by k-means/medoids clustering.*

30.8.1 Hypergeometric tests on annotations

The first approach to using annotations to extract biological information is the hypergeometric annotation test. This test measures the extent to which the annotation categories of features in a smaller gene list, 'A', are over or under-represented relative to those of the features in larger gene list 'B', of which 'A' is a sub-list. Gene list B is often the features of the full experiment, possibly with features which are thought to represent only noise, filtered away. Gene list A is a sub-experiment of the full experiment where most features have been filtered away and only those that seem of interest are kept. Typically gene list A will consist of a list of candidate differentially expressed genes. This could be the gene list obtained after carrying out a statistical analysis on the experiment, and choosing to keep only those features with FDR corrected p-values <0.05 and a fold change larger than 2 in absolute value. The hyper geometric test procedure implemented is similar to the unconditional GOstats test of [Falcon and Gentleman, 2007].

Toolbox | **Microarray and Small RNA Analysis (**  **)**| **Annotation Test** | **Hypergeometric Tests on Annotations (****)**

This will show a dialog where you can select the two experiments - the larger experiment, e.g. the original experiment including the full list of features - and a sub-experiment (see how to create a

sub-experiment in section 30.2.2).

Click **Next**. This will display the dialog shown in figure 30.79.

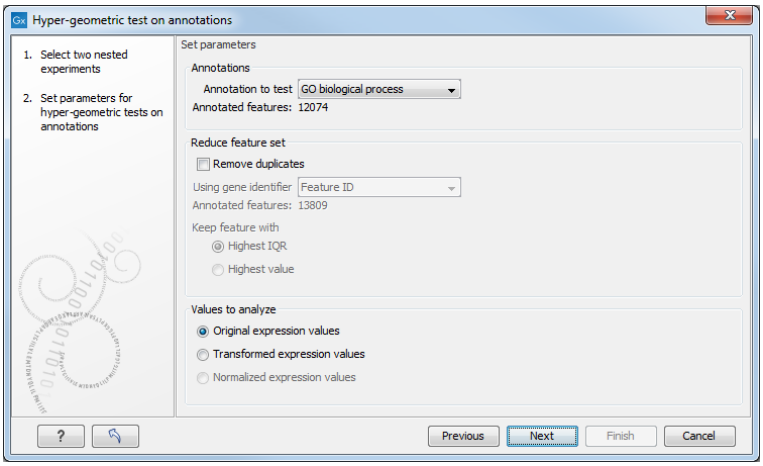


Figure 30.79: *Parameters for performing a hypergeometric test on annotations.*

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 30.4.1). Only features that have a numerical value assigned to them will be used for the analysis. That is, any feature which has a value of plus infinity, minus infinity or NaN will not be included in the feature list taken into the test. Thus, the choice of value at this step can affect the features that are taken forward into the test in two ways:

- If there are features with values of plus infinity, minus infinity or NaN, those features will not be taken forward into the test. This can be a consideration when choosing transformed values, where the mathematical manipulations involved may lead to such values.

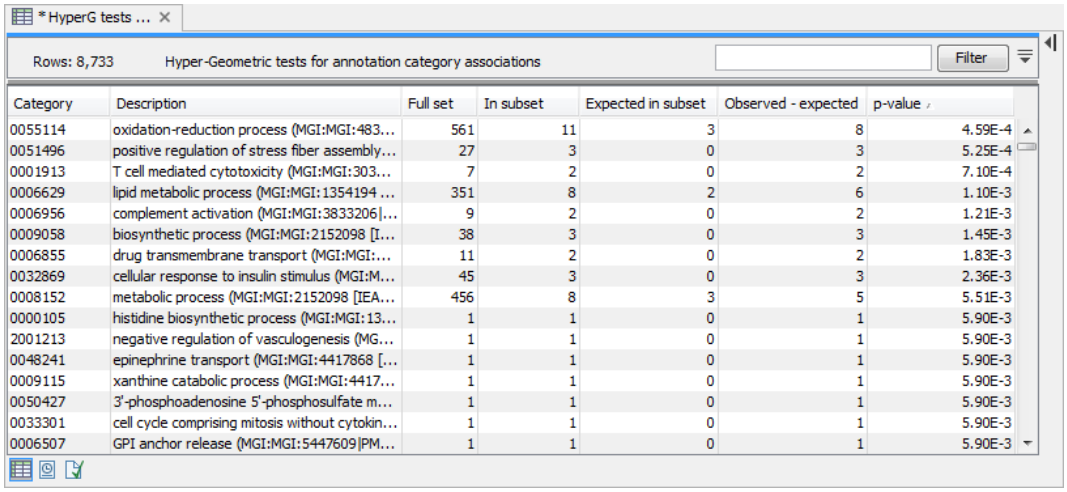
- If you chose to remove duplicates, then the value type you choose here is the value used for checking the highest IQR or value to determine which feature is taken forward into the test.

The final number of features used for the test is reported in this history view of the test results.

Click **Finish** to start the tool.

Result of hypergeometric tests on annotations

The result of performing hypergeometric tests on annotations using GO biological process is shown in figure 30.80.



Category	Description	Full set	In subset	Expected in subset	Observed - expected	p-value
0055114	oxidation-reduction process (MGI:MGI:483...	561	11	3	8	4.59E-4
0051496	positive regulation of stress fiber assembly...	27	3	0	3	5.25E-4
0001913	T cell mediated cytotoxicity (MGI:MGI:303...	7	2	0	2	7.10E-4
0006629	lipid metabolic process (MGI:MGI:1354194 ...	351	8	2	6	1.10E-3
0006956	complement activation (MGI:MGI:3833206|...	9	2	0	2	1.21E-3
0009058	biosynthetic process (MGI:MGI:2152098 [I...	38	3	0	3	1.45E-3
0006855	drug transmembrane transport (MGI:MGI:...	11	2	0	2	1.83E-3
0032869	cellular response to insulin stimulus (MGI:M...	45	3	0	3	2.36E-3
0008152	metabolic process (MGI:MGI:2152098 [IEA...	456	8	3	5	5.51E-3
0000105	histidine biosynthetic process (MGI:MGI:13...	1	1	0	1	5.90E-3
2001213	negative regulation of vasculogenesis (MG...	1	1	0	1	5.90E-3
0048241	epinephrine transport (MGI:MGI:4417868 [...	1	1	0	1	5.90E-3
0009115	xanthine catabolic process (MGI:MGI:4417...	1	1	0	1	5.90E-3
0050427	3'-phosphoadenosine 5'-phosphosulfate m...	1	1	0	1	5.90E-3
0033301	cell cycle comprising mitosis without cytokin...	1	1	0	1	5.90E-3
0006507	GPI anchor release (MGI:MGI:5447609|PM...	1	1	0	1	5.90E-3

Figure 30.80: *The result of testing on GO biological process.*

The table shows the following information:

- **Category**. This is the identifier for the category.
- **Description**. This is the description belonging to the category. Both of these are simply extracted from the annotations.
- **Full set**. The number of features in the original experiment (not the subset) with this category. (Note that this is after removal of duplicates).
- **In subset**. The number of features in the subset with this category. (Note that this is after removal of duplicates).
- **Expected in subset**. The number of features we would have expected to find with this annotation category in the subset, if the subset was a random draw from the full set.
- **Observed - expected**. 'In subset' - 'Expected in subset'
- **p-value**. The tail probability of the hyper geometric distribution This is the value used for sorting the table.

Categories with small p-values are over-represented on the features in the subset relative to the full set.

30.8.2 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** | **Microarray and Small RNA Analysis (**  **)**| **Annotation Test** | **Gene Set Enrichment Analysis (GSEA) (**  **)**

Select an experiment and click **Next**.

Click **Next**. This will display the dialog shown in figure 30.81.

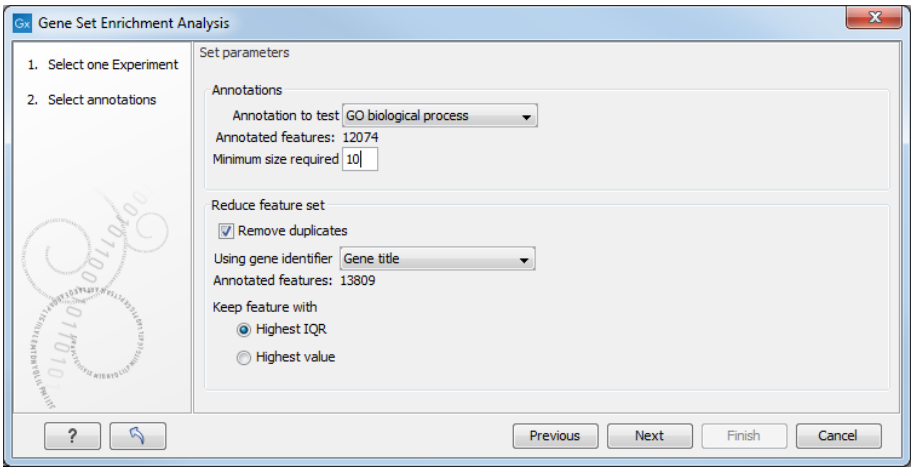


Figure 30.81: *Gene set enrichment analysis on GO biological process.*

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 30.82.

At the top, you can select which values to analyze (see section 30.4.1).

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

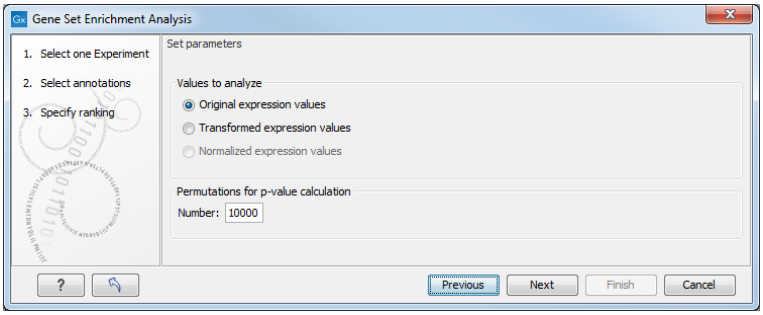


Figure 30.82: *Gene set enrichment analsysis parameters.*

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 30.83.

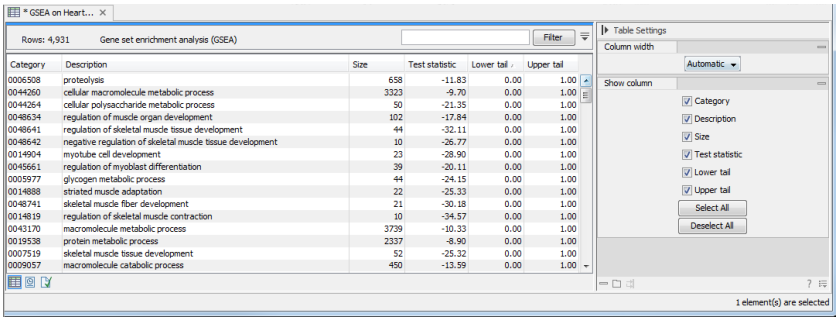


Figure 30.83: *The result of gene set enrichment analysis on GO biological process.*

The table shows the following information:

- **Category**. This is the identifier for the category.
- **Description**. This is the description belonging to the category. Both of these are simply extracted from the annotations.
- **Size**. The number of features with this category. (Note that this is after removal of duplicates).
- **Test statistic**. This is the GSEA test statistic.
- **Lower tail**. This is the mass in the permutation based p-value distribution below the value of the test statistic.
- **Upper tail**. This is the mass in the permutation based p-value distribution above the value of the test statistic.

A small lower (or upper) tail p-value for an annotation category is an indication that features in this category viewed as a whole are perturbed among the groups in the experiment considered.

30.9 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.

30.9.1 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 | **Microarray and Small RNA Analysis (**)| **General Plots** | **Create Histogram** ( )

Select a number of samples (( ), ( ), ( )) or a graph track. When you have selected more than one sample, a histogram will be created for each one. Clicking **Next** will display a dialog as shown in figure 30.84.

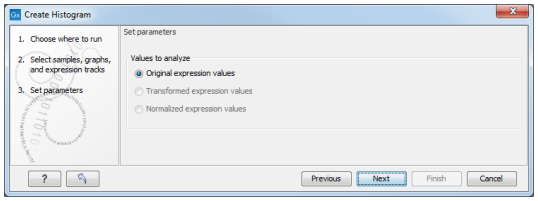


Figure 30.84: *Selecting which values the histogram should be based on.*

In this dialog, you select the values to be used for creating the histogram (see section 30.4.1).

Click **Finish** to start the tool.

Viewing histograms

The resulting histogram is shown in a figure 30.85

The histogram shows the expression value on the x axis (in the case of figure 30.85 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.

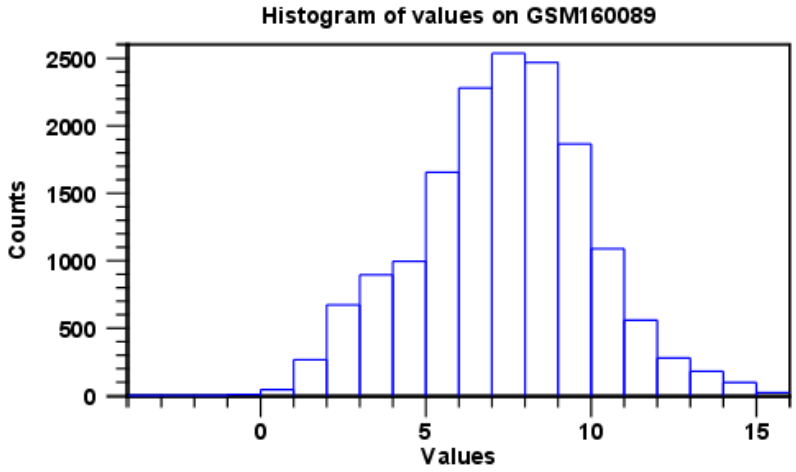


Figure 30.85: *Histogram showing the distribution of transformed expression values.*

  - Outside
  - Inside

- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Break points**. Determines where the bars in the histogram should be:
  - **Sturges method**. This is the default. The number of bars is calculated from the range of values by Sturges formula [Sturges, 1926].
  - **Equi-distanced bars**. This will show bars from **Start** to **End** and with a width of **Sep**.
  - **Number of bars**. This will simply create a number of bars starting at the lowest value and ending at the highest value.

Below the graph preferences, you find **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a principal component plot, you need to save the settings of the **Side Panel** (see section 4.6).

Besides the histogram view itself, the histogram can also be shown in a table, summarizing key properties of the expression values. An example is shown in figure 30.86.

The table lists the following properties:

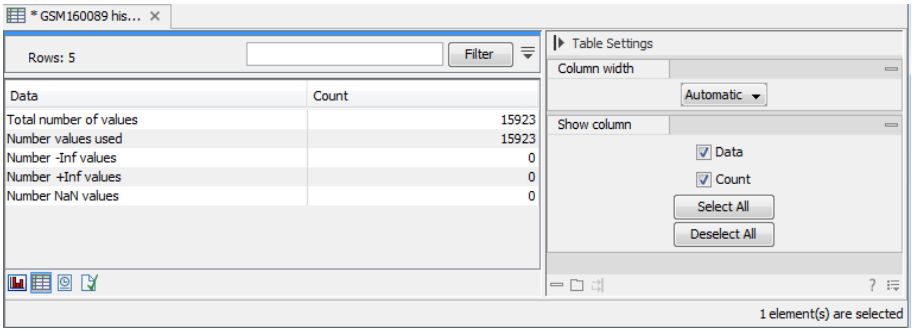


Figure 30.86: *Table view of a histogram.*

- **Number +Inf values**
- **Number -Inf values**
- **Number NaN values**
- **Number values used**
- **Total number of values**

30.9.2 MA plot

The MA plot is a scatter rotated by $45°$. For two samples of expression values it plots for each gene the difference in expression against the mean expression level. MA plots are often used for quality control, in particular, to assess whether normalization and/or transformation is required.

You can create an MA plot comparing two samples:

> **Toolbox** | **Microarray and Small RNA 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 30.87.

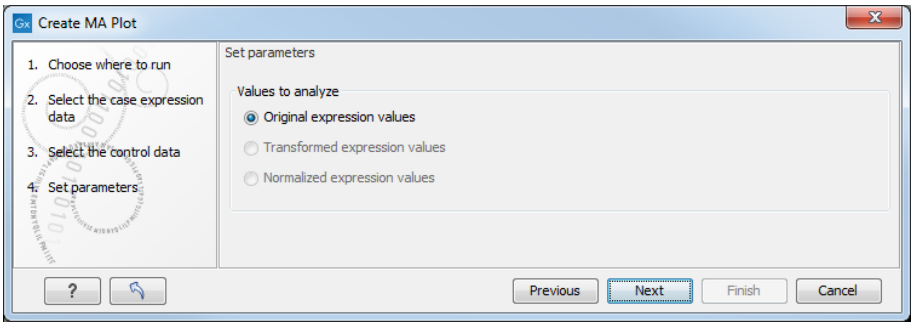


Figure 30.87: *Selecting which values the MA plot should be based on.*

In this dialog, you select the values to be used for creating the MA plot (see section 30.4.1).

Click **Finish** to start the tool.

Viewing MA plots

The resulting plot is shown in a figure 30.88.

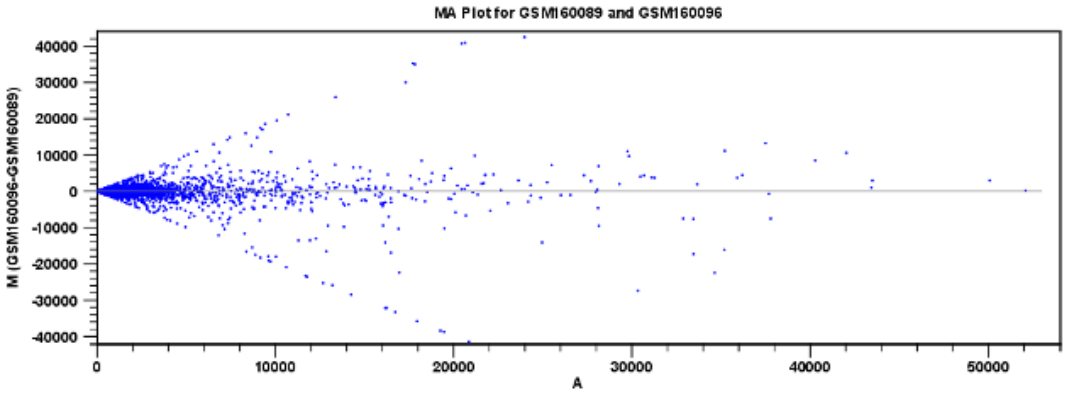


Figure 30.88: *MA plot based on original expression values.*

The X axis shows the mean expression level of a feature on the two samples and the Y axis shows the difference in expression levels for a feature on the two samples. From the plot shown in figure 30.88 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 30.4.2).

Figure 30.89 shows the same two samples where the MA plot has been created using log2 transformed values.

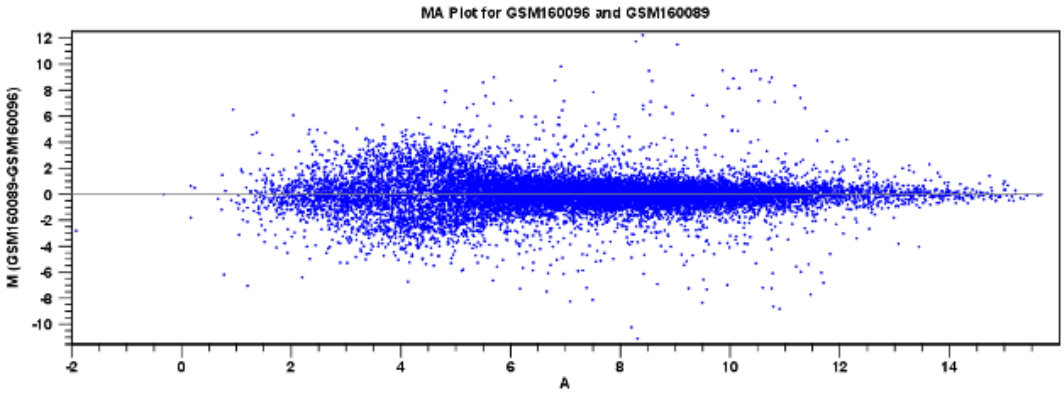


Figure 30.89: *MA plot based on transformed expression values.*

The much more symmetric and even spread indicates that the dependance of the variance on the mean is not as strong as it was before transformation.

In the **Side Panel** to the left, there is a number of options to adjust the view. Under **Graph preferences**, you can adjust the general properties of the plot.

- **Lock axes** This will always show the axes even though the plot is zoomed to a detailed level.
- **Frame** Shows a frame around the graph.

- **Show legends** Shows the data legends.
- **Tick type** Determine whether tick lines should be shown outside or inside the frame.
  - Outside
  - Inside
- **Tick lines at** Choosing Major ticks will show a grid behind the graph.
  - None
  - Major ticks
- **Horizontal axis range** Sets the range of the horizontal axis (x axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **Vertical axis range** Sets the range of the vertical axis (y axis). Enter a value in **Min** and **Max**, and press Enter. This will update the view. If you wait a few seconds without pressing Enter, the view will also be updated.
- **y = 0 axis**. Draws a line where y = 0. Below there are some options to control the appearance of the line:
  - **Line width**
    - Thin
    - Medium
    - Wide
  - **Line type**
    - None
    - Line
    - Long dash
    - Short dash
  - **Line color.** Allows you to choose between many different colors. Click the color box to select a color.
- **Line width**
  - Thin
  - Medium
  - Wide
- **Line type**
  - None
  - Line
  - Long dash
  - Short dash
- **Line color.** Allows you to choose between many different colors. Click the color box to select a color.

Below the general preferences, you find the **Dot properties** preferences, where you can adjust coloring and appearance of the dots:

- **Dot type**
  - None
  - Cross
  - Plus
  - Square
  - Diamond
  - Circle
  - Triangle
  - Reverse triangle
  - Dot
- **Dot color.** Allows you to choose between many different colors. Click the color box to select a color.

Note that if you wish to use the same settings next time you open a scatter plot, you need to save the settings of the **Side Panel** (see section 4.6).

30.9.3 Scatter plot

As described in section 30.2.4, an experiment can be viewed as a scatter plot. However, you can also create a "stand-alone" scatter plot of two samples:

> **Toolbox** | **Microarray and Small RNA 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 30.90.

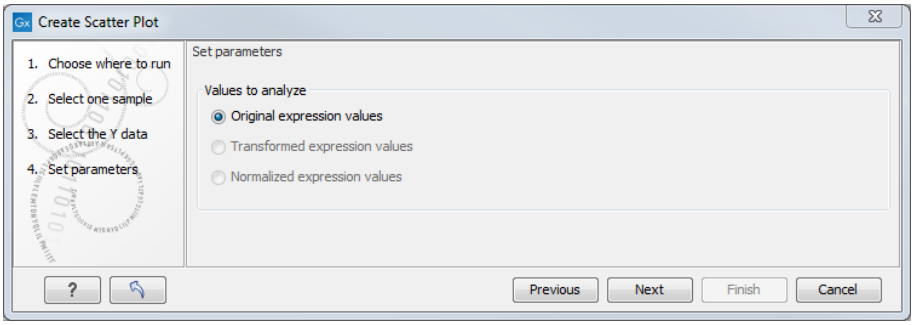


Figure 30.90: *Selecting which values the scatter plot should be based on.*

In this dialog, you select the values to be used for creating the scatter plot (see section 30.4.1).

Click **Finish** to start the tool.

For more information about the scatter plot view and how to interpret it, please see section 30.2.4.

Chapter 31

Helper tools

Contents

31.1 Extract sequences

This tool allows the extraction of sequences from other types of data in the Workbench, such as sequence lists or alignments. The data types you can extract sequences from are:

- Alignments ()
- BLAST result ()
- BLAST overview tables ()
- sequence lists ()
- Contigs and read mappings ()
- Read mapping tables ()
- Read mapping tracks ()
- RNA-Seq mapping results ()

Note! When the Extract Sequences tool is run via the Workbench toolbox on an entire file of one of the above types, **all** sequences are extracted from the data used as input. If only a **subset** of the sequences is desired, for example, the reads from just a small area of a mapping, or the sequences for only a few blast results, then a data set containing just this subsection or subset should be created and the Extract Sequences tool should be run on that.

For extracting a subset of a mapping, please see section 33.7.5 that describes the function "Extract from Selection" that also can be selected from the right click menu (see figure 31.1).

For extracting a subset of a sequence list, you can highlight the sequences of interest in the table view of the sequence list, right click on the selection and launch the Extract Sequences tool.

The Extract Sequences tool can be launched via the Toolbox menu, by going to:

Toolbox | **General Sequence Analysis** ( )| **Extract Sequences** ( )

Alternatively, on all the data types listed above except sequence lists, the option to run this tool appears by right clicking in the relevant area; a row in a table or in the read area of mapping data. An example is shown in figure 31.1.

Please note that for mappings, only the read sequences are extracted. Reference and consensus sequences are not extracted using this tool. Similarly, when extracting sequences from BLAST results, the sequence hits are extracted, not the original query sequence or a consensus sequence.

"Note also, that paired reads will be extracted in accordance with the read group settings, which is specified during the original import of the reads. If the orientation has since been changed (e.g. using the Element Info tab for the sequence list) the read group information will be modified and reads will be extracted as specified by the modified read group. The default read group orientation is forward-reverse."

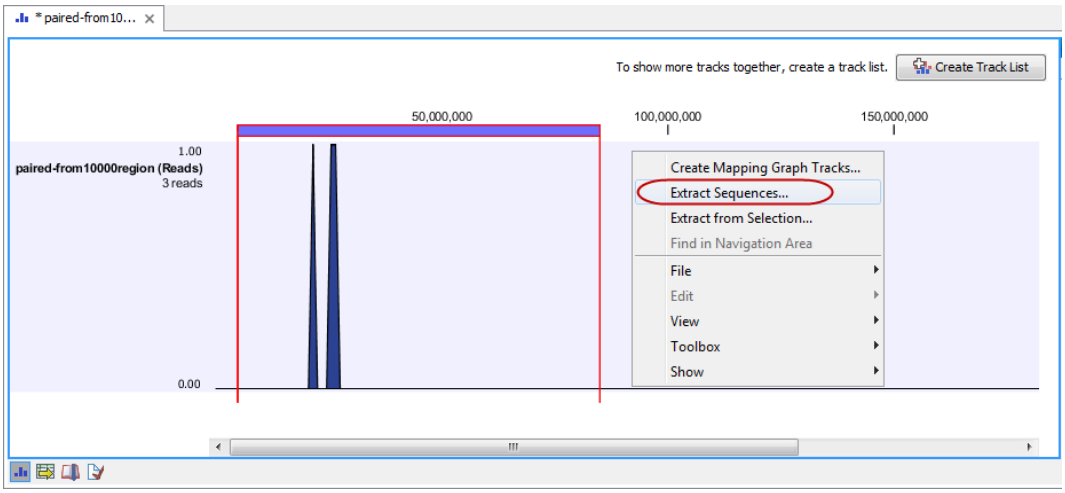


Figure 31.1: *Right click somewhere in the reads track area and select "Extract Sequences".*

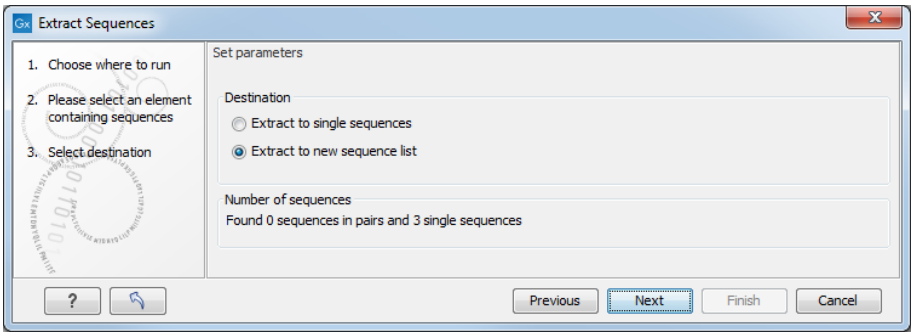


Figure 31.2: *Choosing whether the extracted sequences should be placed in a new list or as single sequences.*

The dialog allows you to select the **Destination**. Here you can choose whether the extracted sequences should be extracted as single sequences or placed in a new sequence list. For most

data types, it will make most sense to choose to extract the sequences into a sequence list. The exception to this is when working with a sequence list, where choosing to extract to a sequence list would create a copy of the same sequence list. In this case, the other option would generally be chosen. This would then result in the generation of individual sequence objects for each sequence in the sequence list.

Below these options, in the dialog, you can see the number of sequences that will be extracted.

31.2 Filter Based on Overlap

The overlap filter will be used for filtering an annotation track based on an overlap with another annotation track. This can be used to e.g. only show variants that fall within genes or regulatory regions or for restricting variants results to only cover a subset of genes.

If you are just interested in finding out whether one particular position overlaps any of the annotations, you can use the advanced table filter and filter on the region column (track tables are described in section 19.3.3). **Toolbox** | **Helper Tools** ( ) | **Filter Based on Overlap** ( )

Select the track you wish to filter and click **Next** to specify the track of overlapping annotations (see figure 31.3).

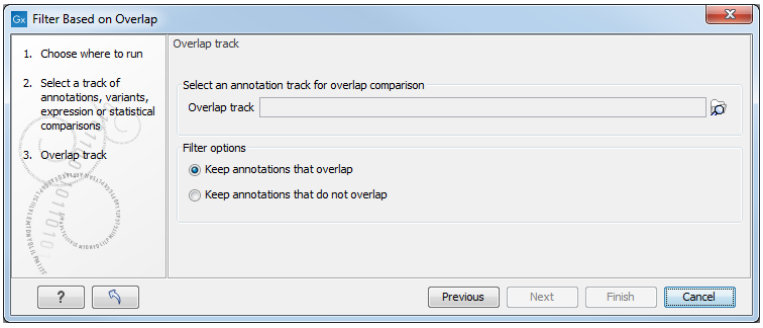


Figure 31.3: *Select overlapping annotations track.*

Next, select the track that should be used for comparison and tick whether you wish to keep annotations that overlap, or whether to keep annotations that do not overlap with the track selected. An overlap has a simple definition – if the annotation used as input has at least one shared position with the other track, there is an overlap. The boundaries of the annotations do not need to match.

Part IX

Cloning

Chapter 32

Cutting and cloning

Contents

Biomedical Genomics Workbench offers graphically advanced *in silico* cloning and design of vectors, together with restriction enzyme analysis and functionalities for managing lists of restriction enzymes.

32.1 Restriction site analyses

There are two ways of finding and showing restriction sites:

- In many cases, the dynamic restriction sites found in the **Side Panel** of sequence views is the fastest and easiest way of showing restriction sites.
- In the **Toolbox** you will find the Cloning and Restriction Sites tool that provides more control on the analysis, and gives you more output options such as a table of restriction sites. It also allows you to perform the same restriction map analysis on several sequences in one step.

32.1.1 Dynamic restriction sites

If you open a sequence, a sequence list etc, you will find a **Restriction Sites** section in the Side Panel.

Restriction sites can be shown on the sequence as colored triangles and lines (figure 32.1): check the "Show" option on top of the Restriction sites section, then specify the enzymes that should be displayed.

Figure 32.1: *Showing restriction sites of ten restriction enzymes.*

The color of the restriction enzyme can be changed by clicking the colored box next to the enzyme's name. The name of the enzyme can also be shown next to the restriction site by selecting **Show name flags** above the list of restriction enzymes.

There is also an option to specify how the **Labels** shown be shown:

- **No labels**. This will just display the cut site with no information about the name of the enzyme. Placing the mouse button on the cut site will reveal this information as a tool tip.
- **Flag**. This will place a flag just above the sequence with the enzyme name (see an example in figure 32.2). Note that this option will make it hard to see when several cut sites are located close to each other. In the circular view, this option is replaced by the Radial option.

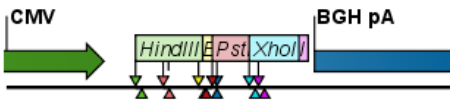


Figure 32.2: *Restriction site labels shown as flags.*

- **Radial**. This option is only available in the circular view. It will place the restriction site labels as close to the cut site as possible (see an example in figure 32.3).

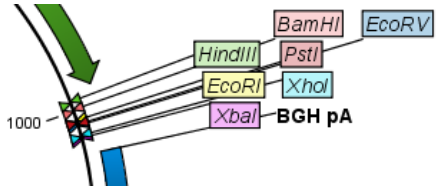


Figure 32.3: *Restriction site labels in radial layout.*

- **Stacked**. This is similar to the flag option for linear sequence views, but it will stack the labels so that all enzymes are shown. For circular views, it will align all the labels on each side of the circle. This can be useful for clearly seeing the order of the cut sites when they are located closely together (see an example in figure 32.4).

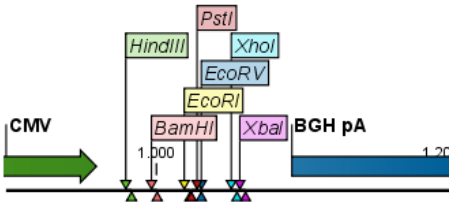


Figure 32.4: *Restriction site labels stacked.*

Note that in a circular view, the **Stacked** and **Radial** options also affect the layout of annotations.

Just above the list of enzymes, three buttons can be used for sorting the list (see figure 32.5).

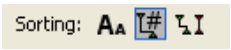


Figure 32.5: *Buttons to sort restriction enzymes.*

- **Sort enzymes alphabetically** (). Clicking this button will sort the list of enzymes alphabetically.
- **Sort enzymes by number of restriction sites** (). This will divide the enzymes into four groups:
  - Non-cutters.
  - Single cutters.
  - Double cutters.
  - Multiple cutters.

  There is a checkbox for each group which can be used to hide / show all the enzymes in a group.
- **Sort enzymes by overhang** (). This will divide the enzymes into three groups:
  - Blunt. Enzymes cutting both strands at the same position.
  - 3'. Enzymes producing an overhang at the 3' end.

– 5’. Enzymes producing an overhang at the 5’ end.

There is a checkbox for each group which can be used to hide / show all the enzymes in a group.

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 32.6.

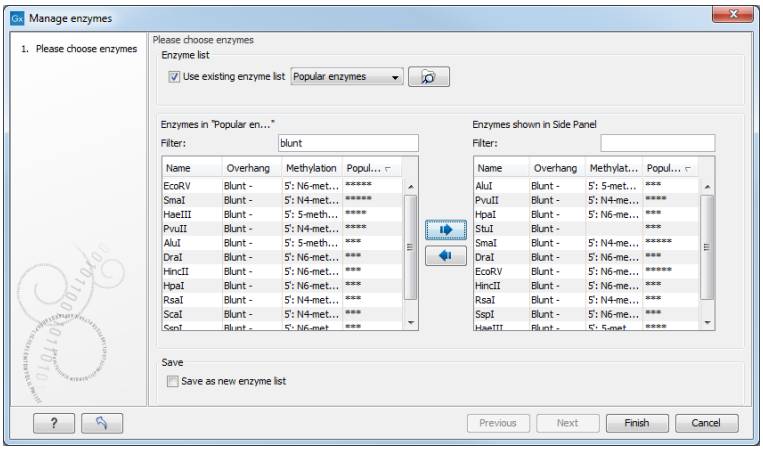


Figure 32.6: *Adding or removing enzymes from the Side Panel.*

At the top, you can choose to **Use existing enzyme list**. Clicking this option lets you select an enzyme list which is stored in the **Navigation Area**. A list of popular enzymes is available in the Example Data folder you can download from the Help menu.

Below there are two panels:

- To the **left**, you can see all the enzymes that are in the list selected above. If you have not chosen to use a specific enzyme list, this panel shows all the enzymes available.
- To the **right**, you can see the list of the enzymes that will be used.

Select enzymes in the left side panel and add them to the right panel by double-clicking or clicking the **Add** button ().

The enzymes can be sorted by clicking the column headings, i.e., Name, Overhang, Methylation or Popularity. This is particularly useful if you wish to use enzymes which produce a 3’ overhang for example.

When looking for a specific enzyme, it is easier to use the Filter. You can type HindIII or blunt into the filter, and the list of enzymes will shrink automatically to only include respectively only the HindIII enzyme, or all enzymes producing a blunt cut.

If you need more detailed information and filtering of the enzymes, you can hover your mouse on an enzyme (see figure 32.7). You can also open a view of an enzyme list saved in the Navigation Area.

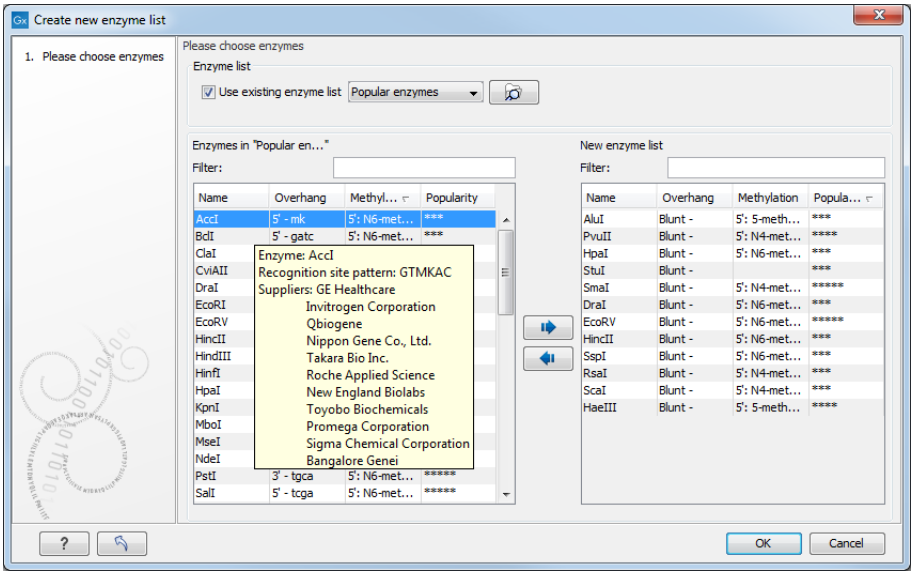


Figure 32.7: *Showing additional information about an enzyme like recognition sequence or a list of commercial vendors.*

At the bottom of the dialog, you can select to save the updated list of enzymes as a new file. When you click **Finish**, the enzymes are added to the Side Panel and the cut sites are shown on the sequence. If you have specified a set of enzymes which you always use, it will probably be a good idea to save the settings in the Side Panel (see section 4.6) for future use.

Show enzymes cutting inside/outside selection

In cases where you have a selection on a sequence, and you wish to find enzymes cutting within the selection but not outside, right-click the selection and choose the option **Show Enzymes Cutting Inside/Outside Selection (**  **)**.

This will open a wizard where you can specify which enzymes should initially be considered (see section 32.1.1). 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 32.8), 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

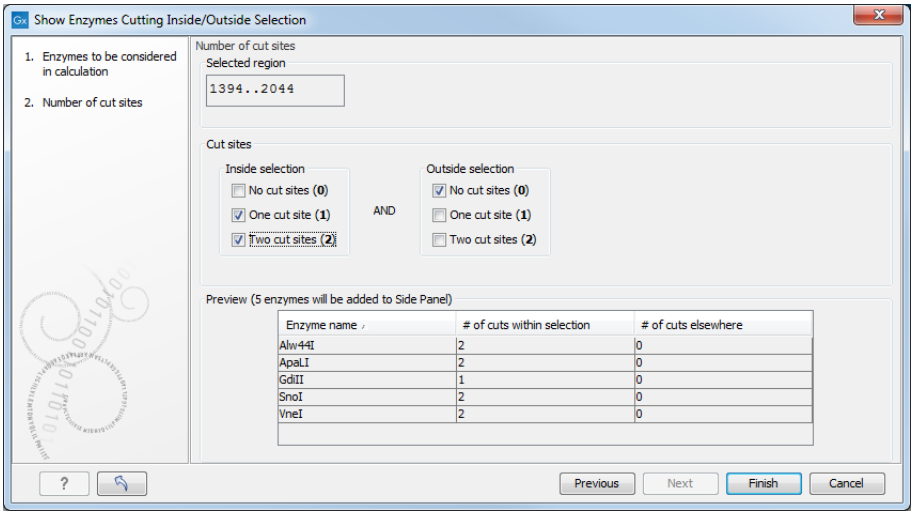


Figure 32.8: *Deciding number of cut sites inside and outside the selection.*

as individual regions. This means that the criteria for cut sites apply to each region.

Show enzymes with compatible ends

A third way of adding enzymes to the Side Panel and thereby displaying them on the sequence is based on the overhang produced by cutting with an enzyme. Right-click on a restriction site and choose to **Show Enzymes with Compatible Ends ( )** to find enzymes producing a compatible overhang (figure 32.9).

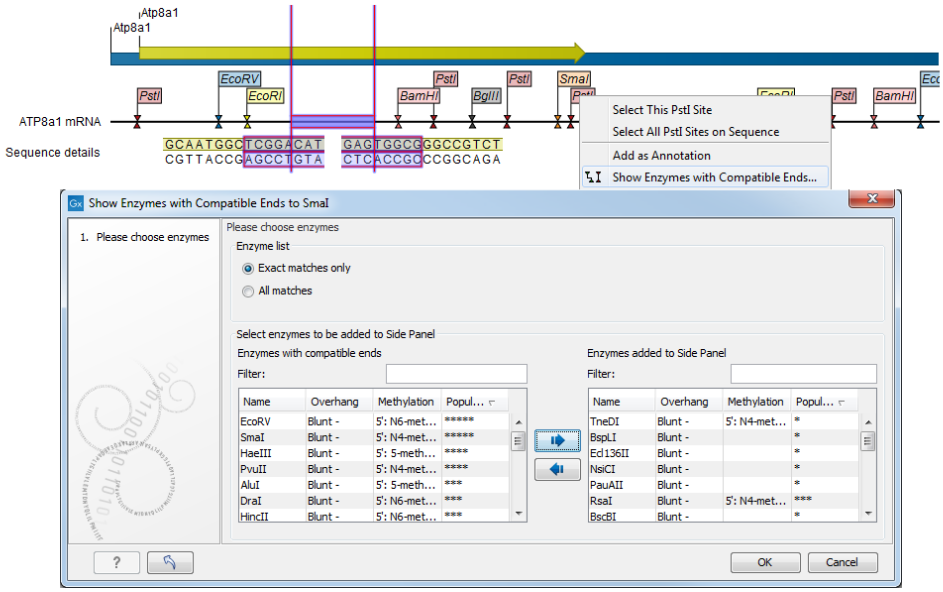


Figure 32.9: *Enzymes with compatible ends.*

At the top you can choose whether the enzymes considered should have an exact match or not. We recommend trying **Exact match** first, and use **All matches** as an alternative if a satisfactory result cannot be achieved. Indeed, since a number of restriction enzymes have ambiguous cut patterns, there will be variations in the resulting overhangs. Choosing **All matches**, you cannot be 100% sure that the overhang will match, and you will need to inspect the sequence further afterwards.

Use the arrows between the two panels to select enzymes which will be displayed on the sequence and added to the Side Panel.

At the bottom of the dialog, the list of enzymes producing compatible overhangs is shown.

When you have added the relevant enzymes, click **Finish**, and the enzymes will be added to the Side Panel and their cut sites displayed on the sequence.

32.1.2 Restriction Site Analysis

Besides the dynamic restriction sites, you can do a more elaborate restriction map analysis with more output format using the Toolbox:

Toolbox | **Cloning and Restriction Sites (**  **)**| **Restriction Site Analysis (**  **)**

You first specify which sequence should be used for the analysis. Then define which enzymes to use as basis for finding restriction sites on the sequence (see section 32.1.1).

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 32.10).

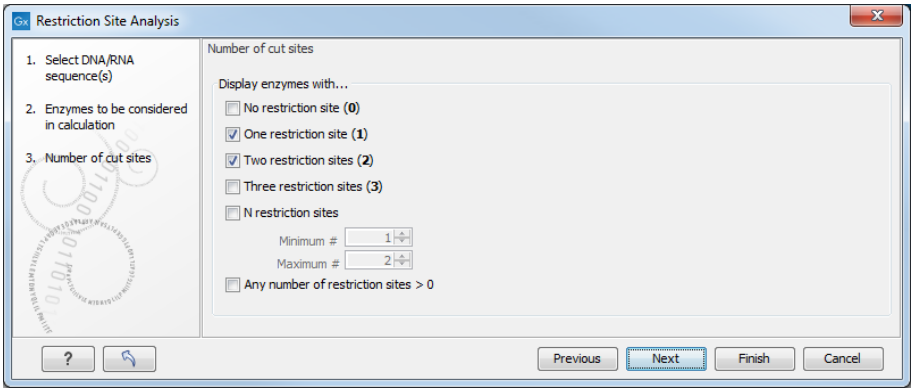


Figure 32.10: *Selecting number of cut sites.*

The default setting is to include the enzymes which cut the sequence one or two times, but you can use the checkboxes to perform very specific searches for restriction sites, for example to find enzymes which do not cut the sequence, or enzymes cutting exactly twice.

The Result handling dialog (figure 32.11) 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 32.12) 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 32.13. If more than one sequence were selected, the table will include the restriction sites of all the sequences. This makes it

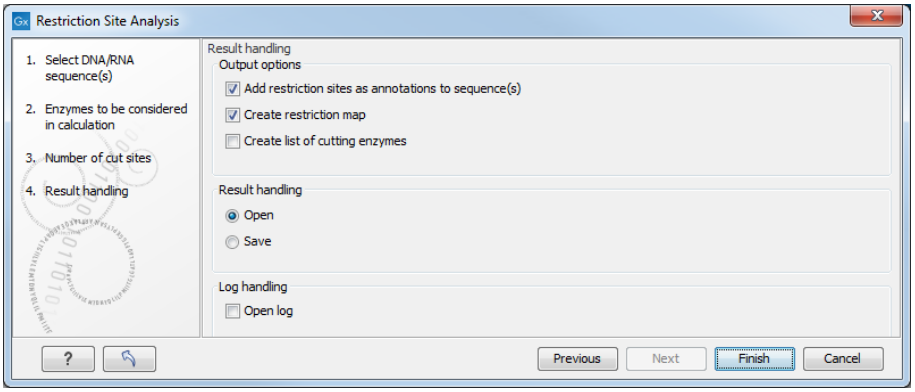


Figure 32.11: *Choosing to add restriction sites as annotations or creating a restriction map.*

Figure 32.12: *The result of the restriction analysis shown as annotations.*

easy to compare the result of the restriction map analysis for two sequences.

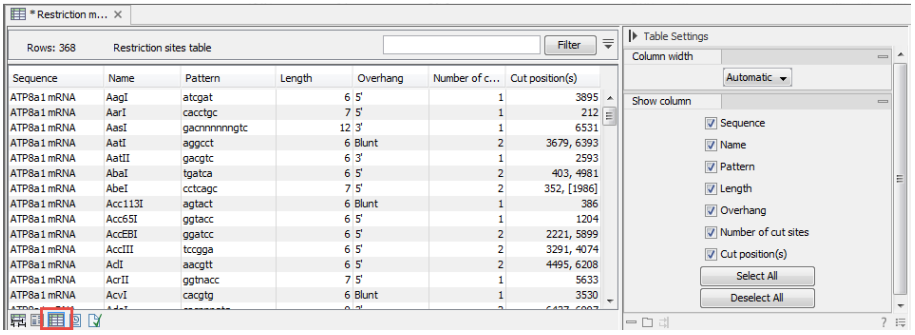


Figure 32.13: *The result of the restriction analysis shown as a table of restriction sites.*

Each row in the table represents a restriction enzyme. The following information is available for each enzyme:

- **Sequence**. The name of the sequence which is relevant if you have performed restriction map analysis on more than one sequence.
- **Name**. The name of the enzyme.
- **Pattern**. The recognition sequence of the enzyme.
- **Length**. the restriction site length.
- **Overhang**. The overhang produced by cutting with the enzyme (3', 5' or Blunt).
- **Number of cut sites**.
- **Cut position(s)**. The position of each cut.
  - **[]** If the enzyme's recognition sequence is on the negative strand, the cut position is put in brackets.
  - **()** Some enzymes cut the sequence twice for each recognition site, and in this case the two cut positions are surrounded by parentheses.

- As a **table of fragments** which shows the sequence fragments that would be the result of cutting the sequence with the selected enzymes (see figure32.14). Click the Fragments button ( ) at the bottom of the view.

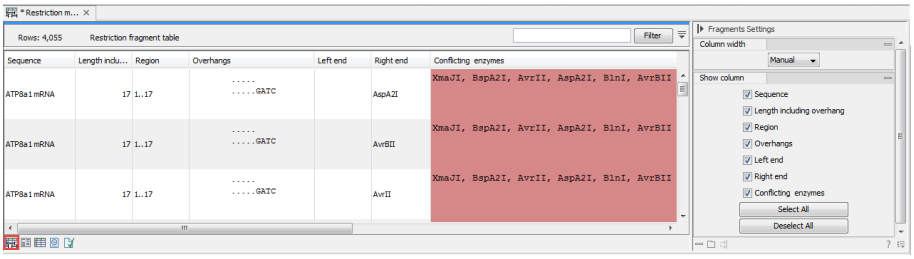


Figure 32.14: *The result of the restriction analysis shown as table of fragments.*

Each row in the table represents a fragment. If more than one enzyme cuts in the same region, or if an enzyme's recognition site is cut by another enzyme, there will be a fragment for each of the possible cut combinations. Furthermore, if this is the case, you will see the names of the other enzymes in the **Conflicting Enzymes** column.

The following information is available for each fragment.

  - **Sequence**. The name of the sequence which is relevant if you have performed restriction map analysis on more than one sequence.
  - **Length including overhang**. The length of the fragment. If there are overhangs of the fragment, these are included in the length (both 3' and 5' overhangs).
  - **Region**. The fragment's region on the original sequence.
  - **Overhangs**. If there is an overhang, this is displayed with an abbreviated version of the fragment and its overhangs. The two rows of dots (.) represent the two strands of the fragment and the overhang is visualized on each side of the dots with the residue(s) that make up the overhang. If there are only the two rows of dots, it means that there is no overhang.
  - **Left end**. The enzyme that cuts the fragment to the left (5' end).
  - **Right end**. The enzyme that cuts the fragment to the right (3' end).
  - **Conflicting enzymes**. If more than one enzyme cuts at the same position, or if an enzyme's recognition site is cut by another enzyme, a fragment is displayed for each possible combination of cuts. At the same time, this column will display the enzymes that are in conflict. If there are conflicting enzymes, they will be colored red to alert the user. If the same experiment were performed in the lab, conflicting enzymes could lead to wrong results. For this reason, this functionality is useful to simulate digestions with complex combinations of restriction enzymes.

  If views of both the fragment table and the sequence are open, clicking in the fragment table will select the corresponding region on the sequence.

- As a **virtual gel** simulation which shows the fragments as bands on a gel (see figure 32.40). For more information about gel electrophoresis, see section 32.5.

32.2 Restriction enzyme lists

Biomedical Genomics Workbench includes all the restriction enzymes available in the **REBASE** database, with methylation shown as performed by the cognate methylase rather than by

Dam/Dcm. If you want to customize the enzyme database for your installation, see section D. However, when performing restriction site analyses, it is often an advantage to use a customized list of enzymes. In this case, the user can create special lists containing for example all enzymes available in the laboratory freezer, or all enzymes used to create a given restriction map or all enzymes that are available form the preferred vendor.

In the example data (see section **??**) 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 *Biomedical Genomics Workbench*.

Create enzyme list *Biomedical Genomics Workbench* uses enzymes from the **REBASE** restriction enzyme database at `http://rebase.neb.com`. If you want to customize the enzyme database for your installation, see section D.

To create an enzyme list of a subset of these enzymes:

File | **New** | **Enzyme list (**  **)**

This opens the dialog shown in figure 32.15

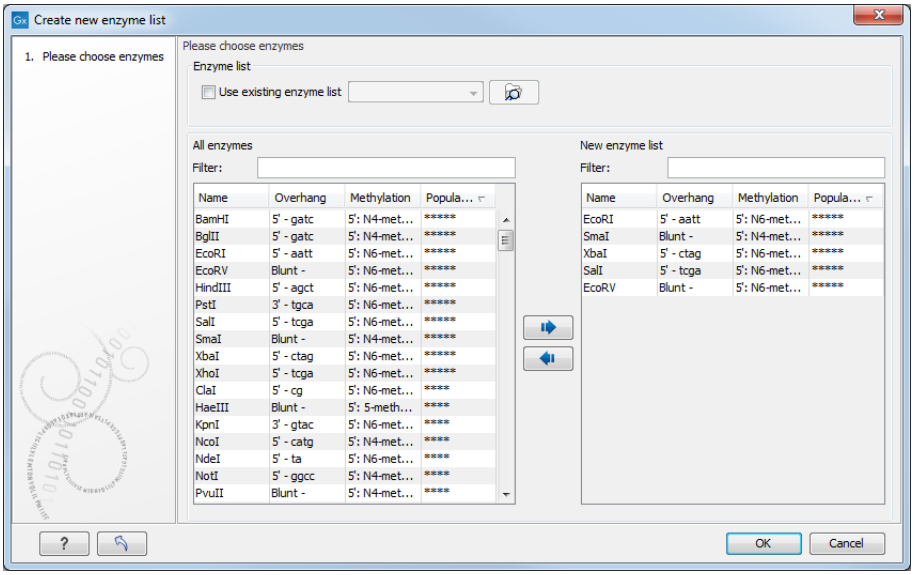


Figure 32.15: *Choosing enzymes for the new enzyme list.*

Choose which enzyme you want to include in the new enzyme list (see section 32.1.1), and click **Finish** to open the enzyme list.

View and modify enzyme list An enzyme list is shown in figure 32.16. It can be sorted by clicking the columns, and you can use the filter at the top right corner to search for specific enzymes, recognition sequences etc.

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 32.15 with the enzyme list shown to the right.

If you wish to extract a subset of an enzyme list, open the list, select the relevant enzymes, right-click on the selection and choose to **Create New Enzyme List from Selection** ().

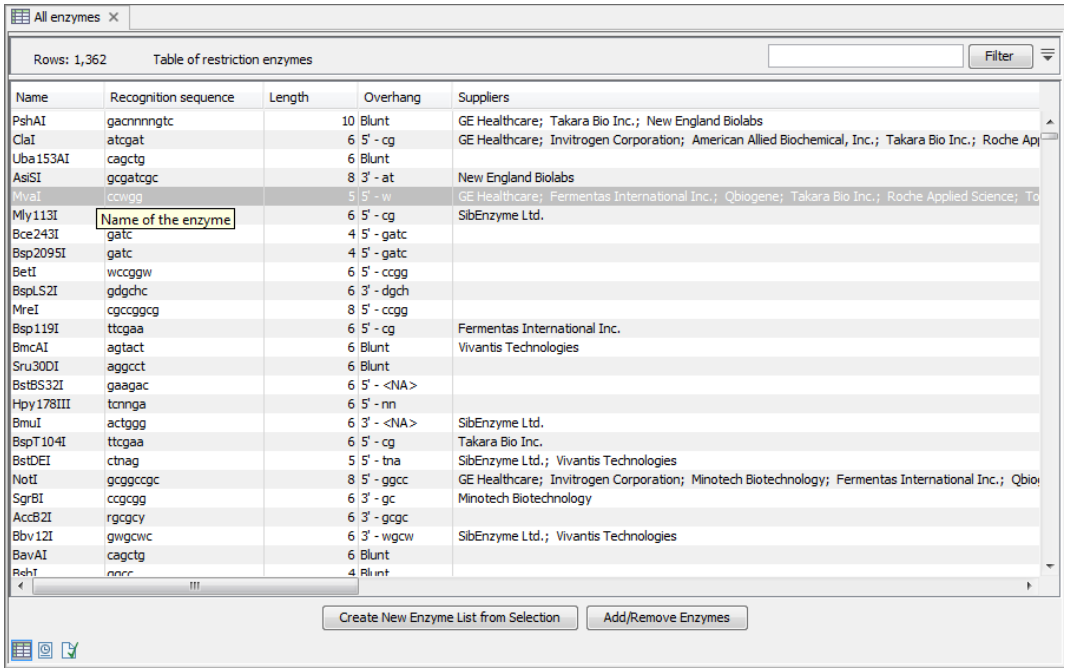


Figure 32.16: *An enzyme list.*

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.

32.3 Molecular cloning

The *in silico* cloning process in *Biomedical Genomics Workbench* begins with the Cloning tool:

Cloning and Restriction Sites ( )| Cloning ( )

This will open a dialog where you can select both the sequences containing the fragments you want to clone, as well as the one to be used as vector (figure 32.17).

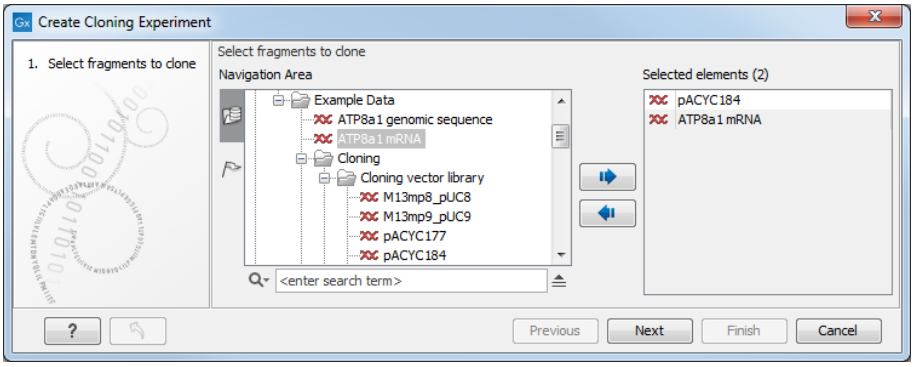


Figure 32.17: *Selecting the sequences containing the fragments you want to clone and the vector.*

Biomedical Genomics Workbench will now create a sequence list of the selected fragments and vector sequences. For cloning work, open the sequence list and switch to the **Cloning** ( ) editor at the bottom of the view (figure 32.18).

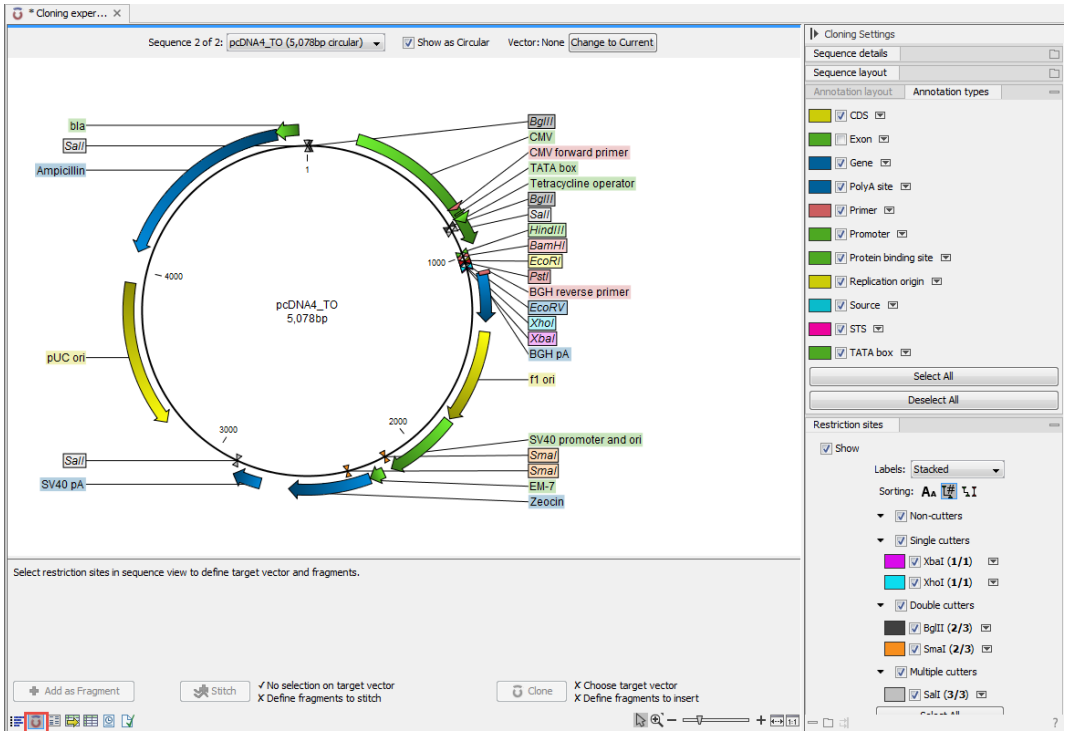


Figure 32.18: *Cloning editor view of the sequence list. Choose which sequence to display from the drop down menu.*

If you later in the process need additional sequences, right-click anywhere on the empty white area of the view and choose to "Add Sequences".

32.3.1 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 10.1). In particular, this means that annotations can be displayed on the sequences to guide the choice of regions to clone.

However, the cloning editor has a special layout with three distinct areas (in addition to the **Side Panel** found in other sequence views as well):

- At the top, there is a panel to switch between the sequences selected as input for the cloning. You can also specify whether the sequence should be visualized **as circular** or as a fragment. On the right-hand side, you can select a vector: the button is by default set to **Change to Current**. Click on it to select the currently shown sequence as **vector**.
- In the middle, the selected sequence is shown. This is the central area for defining how the cloning should be performed.
- At the bottom, there is a panel where the selection of fragments and target vector is performed.

The cloning editor can be activated in different ways. One way is to click on the **Cloning Editor** icon ( ) in the view area when a sequence list has been opened in the sequence list editor. Another way is to create a new cloning experiment (the actual data object will still be a sequence

list) using the **Cloning** ( ) action from the toolbox. Using this action the user collects a set of existing sequences and creates a new sequence list.

The cloning editor can be used in two different ways:

- **The cloning mode**, when the user has selected one of the sequences as 'Vector'. In the cloning mode, the user opens up the vector by applying one or more cuts to the vector, thereby creating an opening for insertion of other sequence fragments. From the remaining sequences in the cloning experiment/sequence list, either complete sequences or fragments created by cutting can be inserted into the vector. In the cloning adapter dialog, the user can switch the order of the inserted fragments and rotate them prior to adjusting the overhangs to match the cloning conditions.
- **The stitch mode**, when the user has not selected a sequence as 'Vector'. In stitch mode, the user can select a number of fragments (either full sequences or cuttings) from the cloning experiment. These fragments can then be stitched together into one single new and longer sequence. In the stitching adapter dialog, the user can switch order and rotate the fragments prior to adjusting the overhangs to match the stitch conditions.

32.3.2 The cloning workflow

The *cloning workflow* is designed to support restriction cloning workflows through the following steps:

1. **Define one or more fragments**

   First, select the sequence containing the cloning fragment in the list at the top of the view. Next, make sure the restriction enzyme you wish to use is listed in the **Side Panel** (see section 32.1.1). 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 32.19).

   In this example you can see that there are now three fragments that can be used for cloning listed in the panel below the view. The fragment selected per default is the one that is in between the cut sites selected.

   If the entire sequence should be selected as fragment, click **Add as Fragment** ( ).

   At any time, the selection of cut sites can be cleared by clicking the **Remove** ( ) icon to the right of the target vector selections.

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 32.20).

   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.

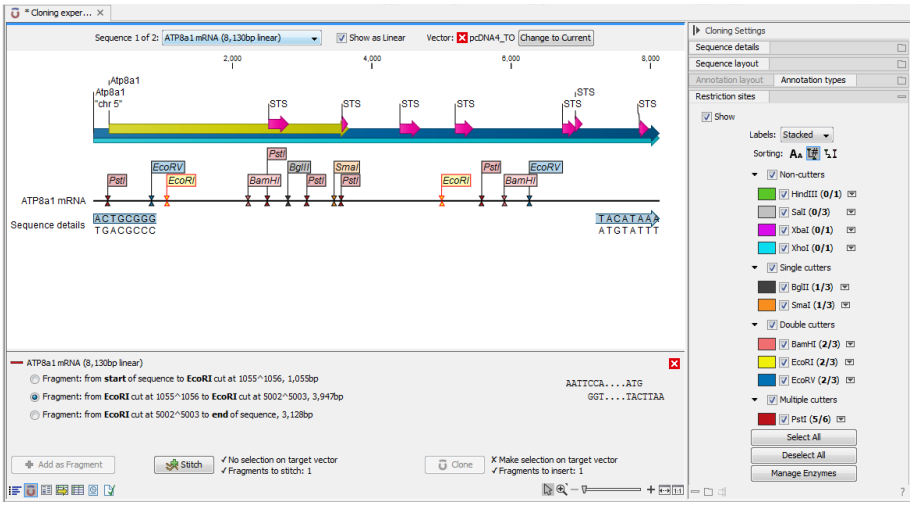


Figure 32.19: *EcoRI cut sites selected to cut out fragment.*

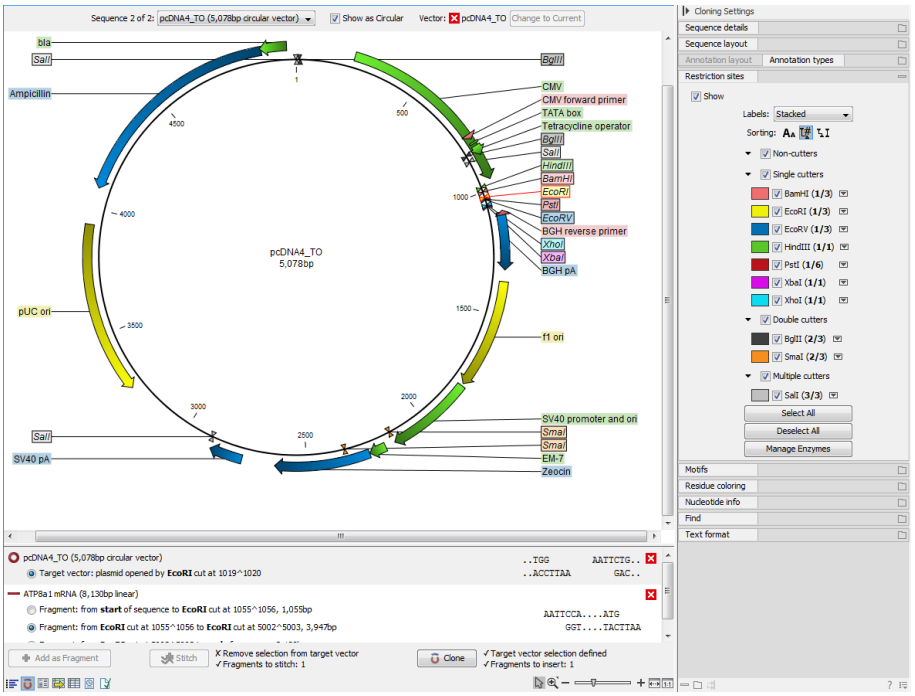


Figure 32.20: *EcoRI site used to open the vector. Note that the "Cloning" button has now been enabled as both criteria ("Target vector selection defined" and "Fragments to insert:...") have been defined.*

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 32.21)

   This dialog visualizes the details of the insertion. The vector sequence is on each side shown in a faded gray color. In the middle the fragment is displayed. If the overhangs of the sequence and the vector do not match ( ), you will not be able to click **Finish**. But you can blunt end or fill in the overhangs using the **drag handles** ( ) until the overhangs match ( ).

   The fragment can be reverse complemented by clicking the **Reverse complement fragment**

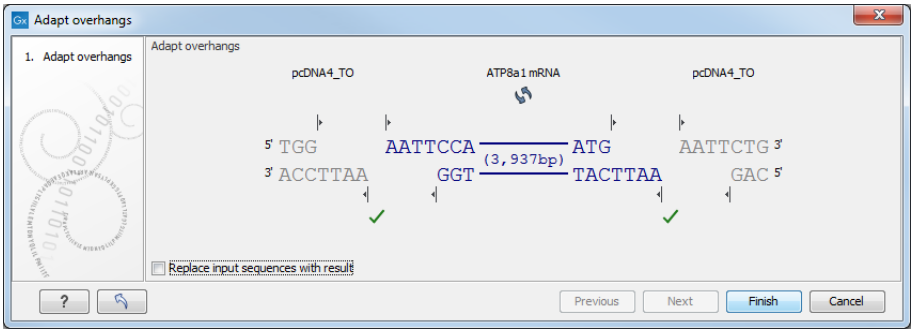


Figure 32.21: *Showing the insertion point of the vector.*

(🔄).

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.

32.3.3 Manual cloning

If you wish to use the manual way of cloning, you still create a sequence list with the Cloning tool, but can skip the "Perform cloning" step of the cloning workflow explained above in section 32.3.2. Instead, all manipulations of sequences are done manually, using right-click menus. These menus have two different appearances depending on where you click, as visualized in figure 32.22.

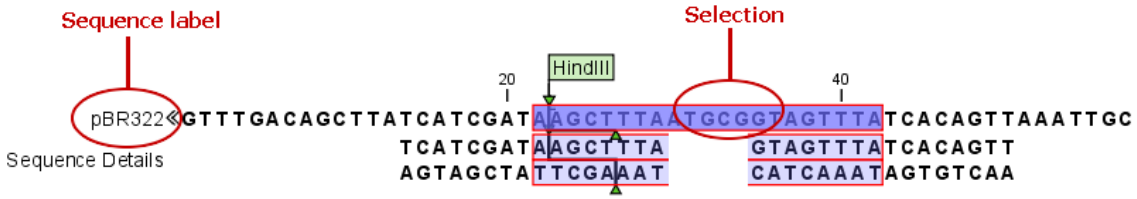


Figure 32.22: *The red circles mark the two places you can use for manipulating the sequences.*

Manipulate the whole sequence

Right-click the sequence label to the left to see the menu shown in figure 32.23.

- **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

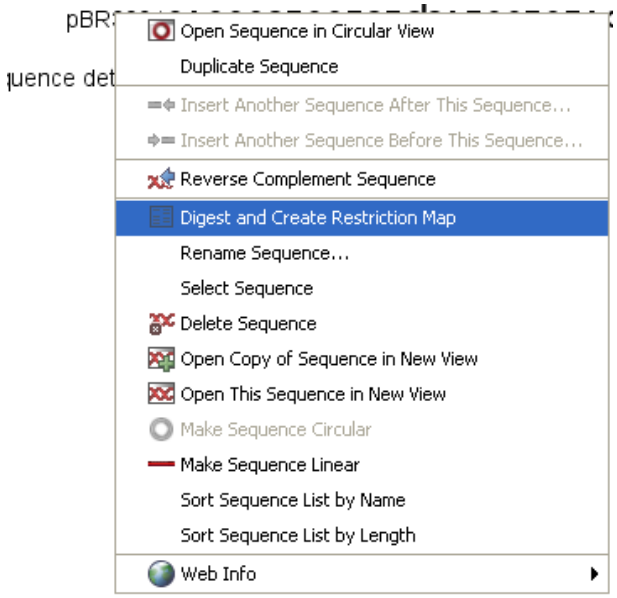


Figure 32.23: *Right click on the sequence in the cloning view.*

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 **??**
- **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 32.24.

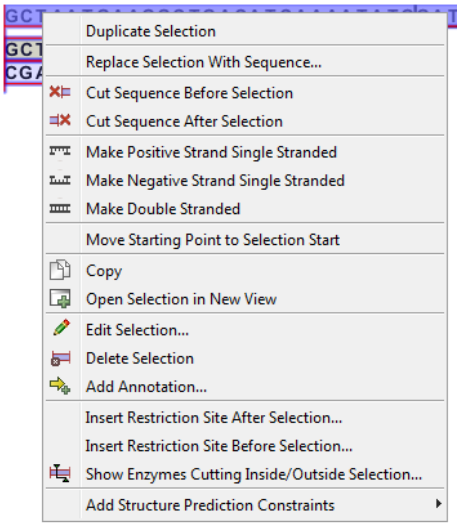


Figure 32.24: *Right click on a sequence selection in the cloning view.*

- **Duplicate Selection.** If a selection on the sequence is duplicated, the selected region will be added as a new sequence to the cloning editor. The new sequence name representing the length of the fragment. When double-clicking on a sequence, the region between the two closest restriction sites is automatically selected.
- **Replace Selection with sequence.** Replaces the selected region with a sequence selected from the drop down menu listing all sequences in the cloning editor.
- **Cut Sequence Before Selection ( ).** Cleaves the sequence before the selection and will result in two smaller fragments.
- **Cut Sequence After Selection ( ).** Cleaves the sequence after the selection and will result in two smaller fragments.
- **Make Positive Strand Single Stranded ( ).** Makes the positive strand of the selected region single stranded.
- **Make Negative Strand Single Stranded ( ).** Makes the negative strand of the selected region single stranded.
- **Make Double Stranded ( ).** This will make the selected region double stranded.
- **Move Starting Point to Selection Start.** This is only active for circular sequences. It will move the starting point of the sequence to the beginning of the selection.
- **Copy ( ).** Copies the selected region to the clipboard, which will enable it for use in other programs.
- **Open Selection in New View ( ).** Opens the selected region in the normal sequence view.
- **Edit Selection ( ).** Opens a dialog box in which is it possible to edit the selected residues.
- **Delete Selection ( ).** Deletes the selected region of the sequence.

- **Add Annotation ( ).** Opens the **Add annotation** dialog box.
- **Insert Restriction Sites After/Before Selection.** Shows a dialog where you can choose from a list restriction enzymes (see section 32.3.4).
- **Show Enzymes Cutting Inside/Outside Selection ( ).** Adds enzymes cutting this selection to the Side Panel.
- **Add Structure Prediction Constraints.** This is relevant for RNA secondary structure prediction:
  - **Force Stem Here** is activated after choosing 2 regions of equal length on the sequence. It will add an annotation labeled "Forced Stem" and will force the algorithm to compute minimum free energy and structure with a stem in the selected region.
  - **Prohibit Stem Here** is activated after choosing 2 regions of equal length on the sequence. It will add an annotation labeled "Prohibited Stem" to the sequence and will force the algorithm to compute minimum free energy and structure without a stem in the selected region.
  - **Prohibit From Forming Base Pairs** will add an annotation labeled "No base pairs" to the sequence, and will force the algorithm to compute minimum free energy and structure without a base pair containing any residues in the selected region.

Insert one sequence into another

Sequences can be inserted into each other in various ways as described in the lists above. When you choose to insert one sequence into another, you will be presented with a dialog where all sequences in the sequence list are present (see figure 32.25).

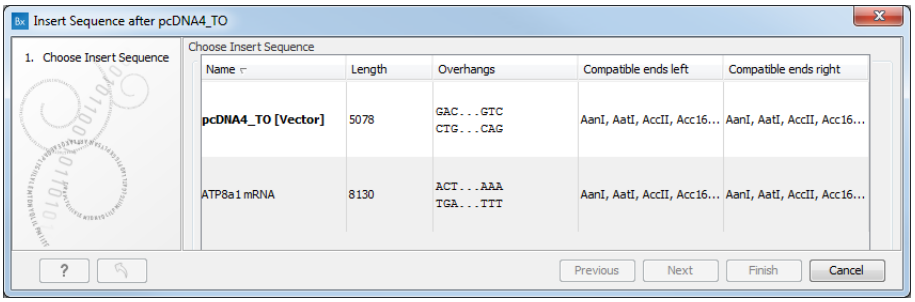


Figure 32.25: *Select a sequence for insertion.*

The sequence that you have chosen to insert into will be marked with **bold** and the text **[vector]** is appended to the sequence name. Note that this is completely unrelated to the vector concept in the cloning workflow described in section 32.3.2.

Furthermore, t he 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 32.26).

At the top is a button to reverse complement the inserted sequence.

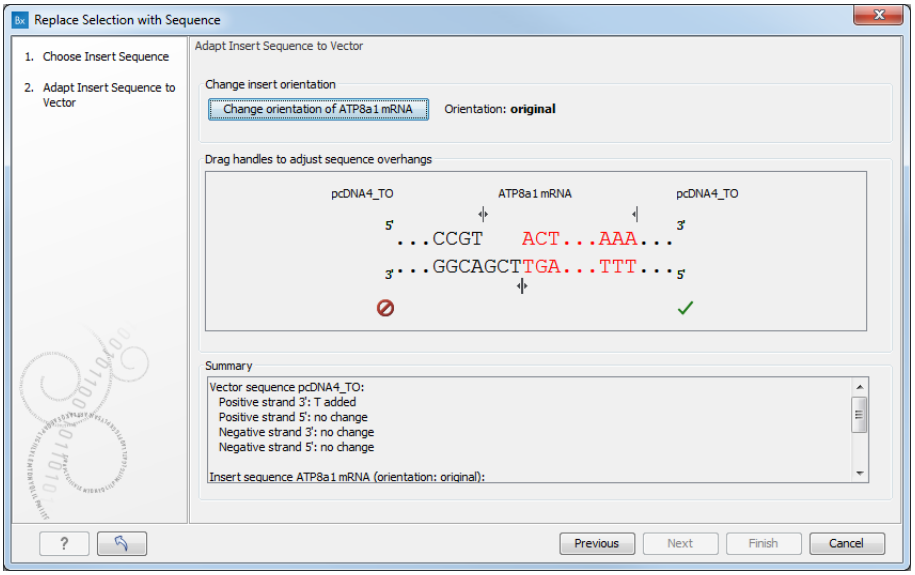


Figure 32.26: *Drag the handles to adjust overhangs.*

Below is a visualization of the insertion details. The inserted sequence is at the middle shown in red, and the vector has been split at the insertion point and the ends are shown at each side of the inserted sequence.

If the overhangs of the sequence and the vector do not match ( ), you can blunt end or fill in the overhangs using the **drag handles** ( ) until it does ( ).

At the bottom of the dialog is a summary field which records all the changes made to the overhangs. This contents of the summary will also be written in the history ( ) of the cloning experiment.

When you click **Finish**, the sequence is inserted and highlighted by being selected.

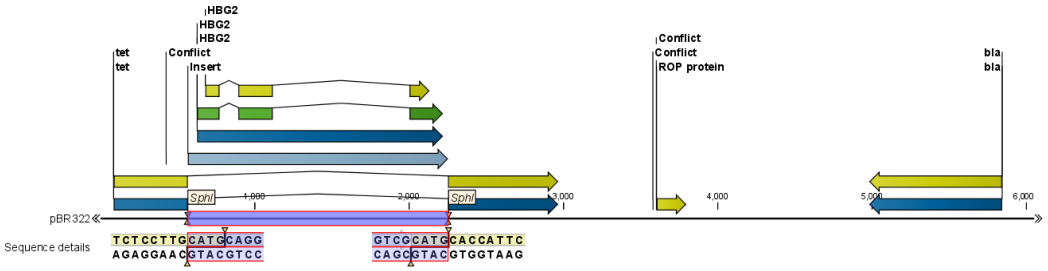


Figure 32.27: *One sequence is now inserted into the cloning vector. The sequence inserted is automatically selected.*

32.3.4 Insert restriction site

If you right-click on a selected region of a sequence, you find this option for inserting the recognition sequence of a restriction enzyme before or after the region you selected. This will display a dialog as shown in figure 32.28

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

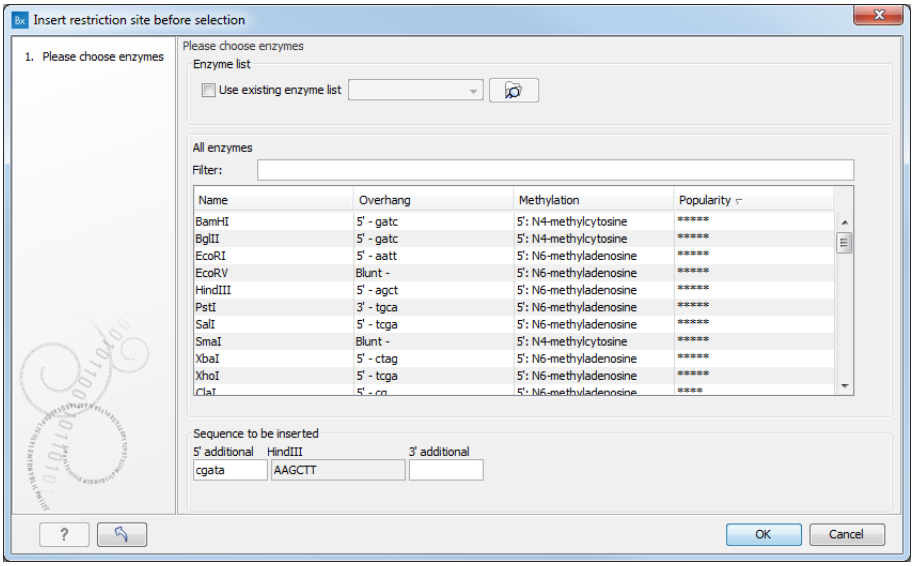


Figure 32.28: *Inserting a restriction site and potentially a recognition sequence.*

(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.

32.4 Gateway cloning

Biomedical Genomics Workbench offers tools to perform *in silico* Gateway cloning[1], including Multi-site Gateway cloning.

The three tools for doing Gateway cloning in the *Biomedical Genomics 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 *Biomedical Genomics 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.

[1]Gateway is a registered trademark of Invitrogen Corporation

32.4.1 Add attB sites

The first step in the Gateway cloning process is to amplify the target sequence with primers including so-called attB sites:

> **Toolbox** | **Cloning and Restriction Sites ( )**| **Gateway Cloning ( )** | **Add attB Sites ( )**

This will open a dialog where you can select one or more sequences. Note that if your fragment is part of a longer sequence, you will need to extract it prior to starting the tool: select the relevant region (or an annotation) of the original sequence, right-click the selection and choose to **Open Annotation in New View**. **Save** ( ) the new sequence in the Navigation Area.

When you have selected your fragment(s), click **Next**.

This will allow you to choose which attB sites you wish to add to each end of the fragment as shown in figure 32.29.

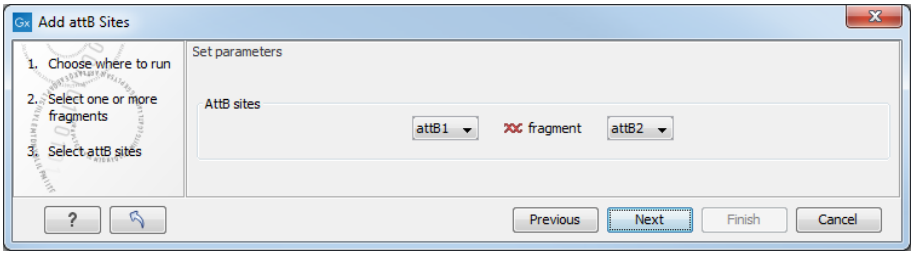


Figure 32.29: *Selecting which attB sites to add.*

The default option is to use the attB1 and attB2 sites. If you have selected several fragments and wish to add different combinations of sites, you will have to run this tool once for each combination.

Next, you are given the option to extend the fragment with additional sequences by extending the primers 5' of the template-specific part of the primer, i.e., between the template specific part and the attB sites.

You can manually type or paste in a sequence of your choice, but it is also possible to click in the text field and press **Shift + F1 (Shift + Fn + F1 on Mac)** to show some of the most common additions (see figure 32.30). Use the up and down arrow keys to select a tag and press **Enter**. To learn how to modify the default list of primer additions, see section 32.4.1.

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.

In the next step, specify the length of the template-specific part of the primers as shown in figure 32.31.

Biomedical Genomics Workbench is not doing any kind of primer design when adding the attB sites. As a user, you simply specify the length of the template-specific part of the primer, and together with the attB sites and optional primer additions, this will be the primer. The primer region will be annotated in the resulting attB-flanked sequence. You can also choose to get a list of primers in the Result handling dialog (see figure 32.32).

The attB sites, the primer additions and the primer regions are annotated in the final result as

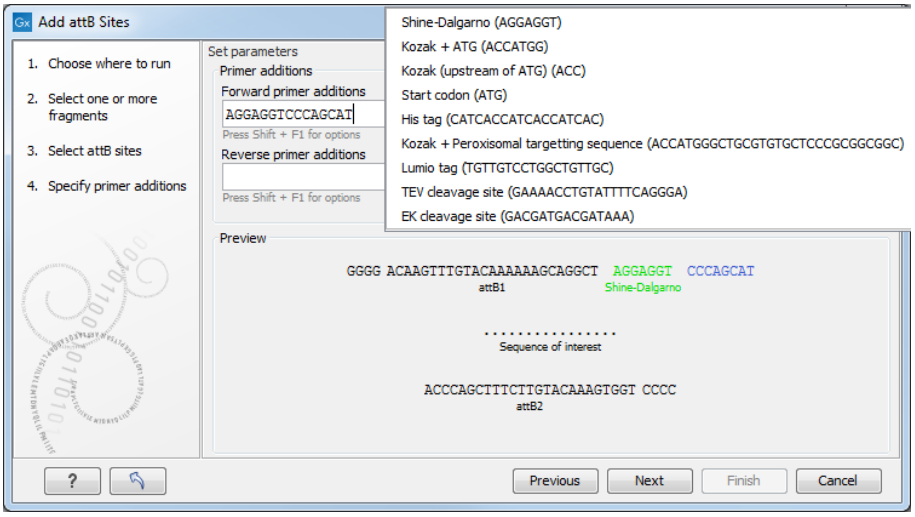


Figure 32.30: *Primer additions 5' of the template-specific part of the primer where a Shine-Dalgarno site has been added between the attB site and the gene of interest.*

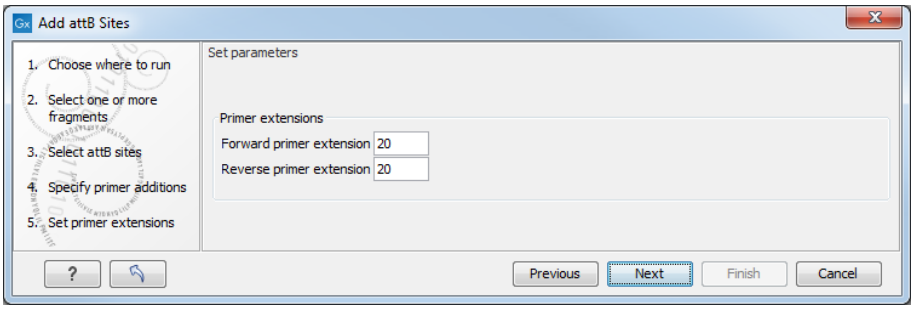


Figure 32.31: *Specifying the length of the template-specific part of the primers.*

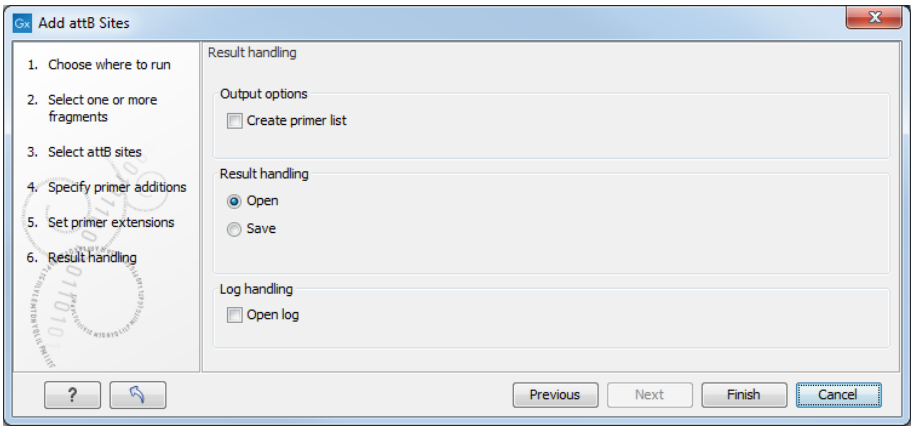


Figure 32.32: *Besides the main output which is a copy of the input sequence(s) now including attB sites and primer additions, you can get a list of primers as output.*

shown in figure 32.33 (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 32.4.2).

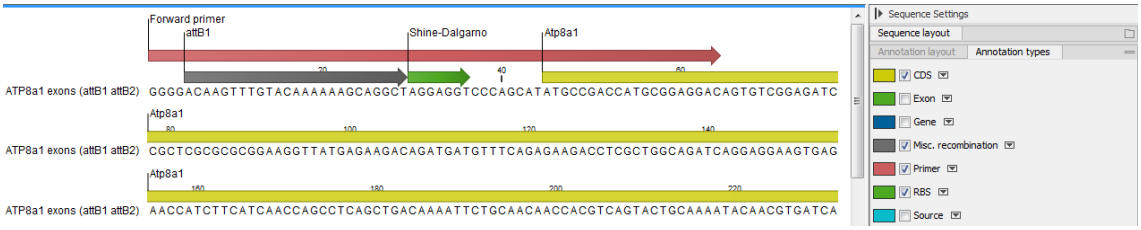


Figure 32.33: *the attB site plus the Shine-Dalgarno primer addition is annotated.*

Extending the pre-defined list of primer additions

The list of primer additions shown when pressing **Shift+F1** (on Mac: Shift + fn + F1) in the dialog shown in figure 32.30 can be configured and extended. If there is a tag that you use a lot, you can add it to the list for convenient and easy access later on. This is done in the **Preferences**:

Edit | **Preferences** | **Data**

In the table **Multisite Gateway Cloning primer additions** (see figure 32.34), select which primer addition options you want to add to forward or reverse primers. You can edit the existing elements in the table by double-clicking any of the cells, or you can use the buttons below to **Add Row** or **Delete Row**. If you by accident have deleted or modified some of the default primer additions, you can press **Add Default Rows**. Note that this will not reset the table but only add all the default rows to the existing rows.

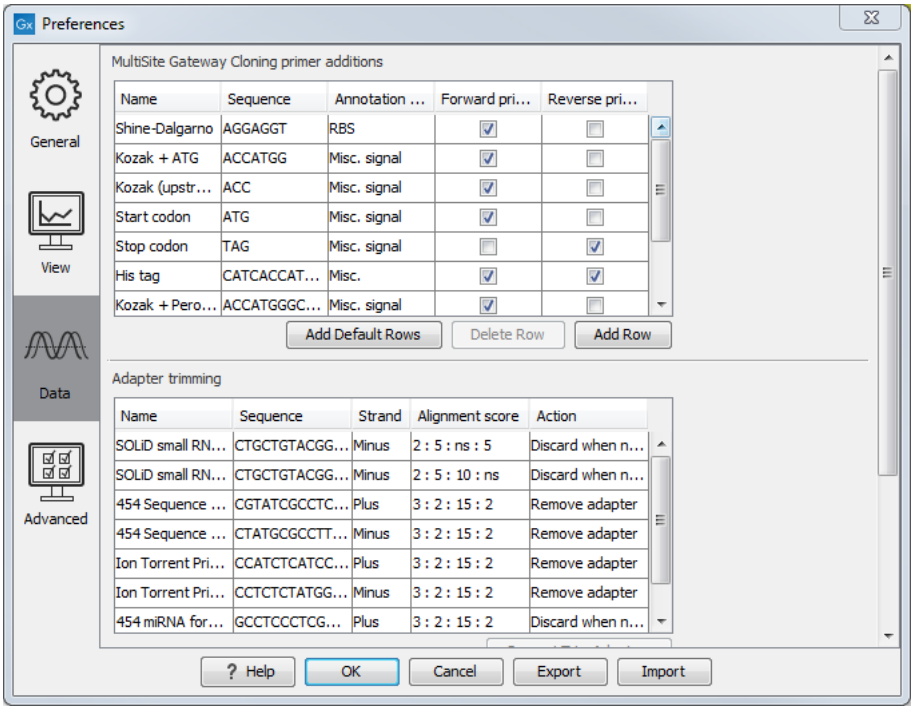


Figure 32.34: *Configuring the list of primer additions available when adding attB sites.*

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.

32.4.2 Create entry clones (BP)

The next step in the Gateway cloning work flow is to recombine the attB-flanked sequence of interest into a donor vector to create an entry clone. Before proceeding to this step, make sure that the sequence of the destination vector was saved in the Navigation Area: find the relevant vector sequence on the Thermo Fisher Scientific's website, copy it, and paste it in in the field that opens when you choose **New | Sequence** in the workbench. Fill in additional information appropriately (enter a "Name", check the "Circular" option) and save the sequence in the Navigation Area.

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

In the first wizard window, select one or more sequences to be recombined into your donor vector. Note that the sequences you select should be flanked with attB sites (see section 32.4.1). You can select more than one sequence as input, and the corresponding number of entry clones will be created.

In the following dialog (figure 32.35), you can specify a donor vector.

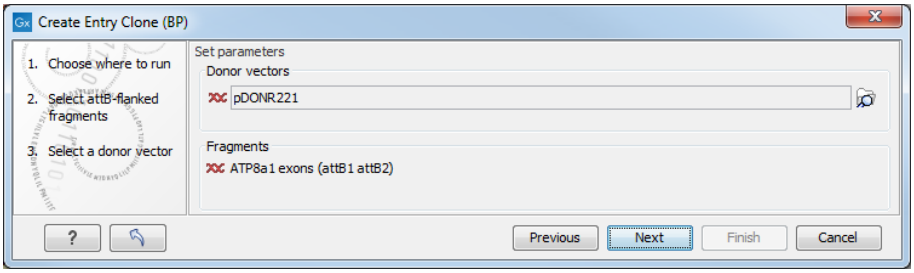


Figure 32.35: *Selecting one or more donor vectors.*

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 32.36.

Note that the bi-product of the recombination is not part of the output.

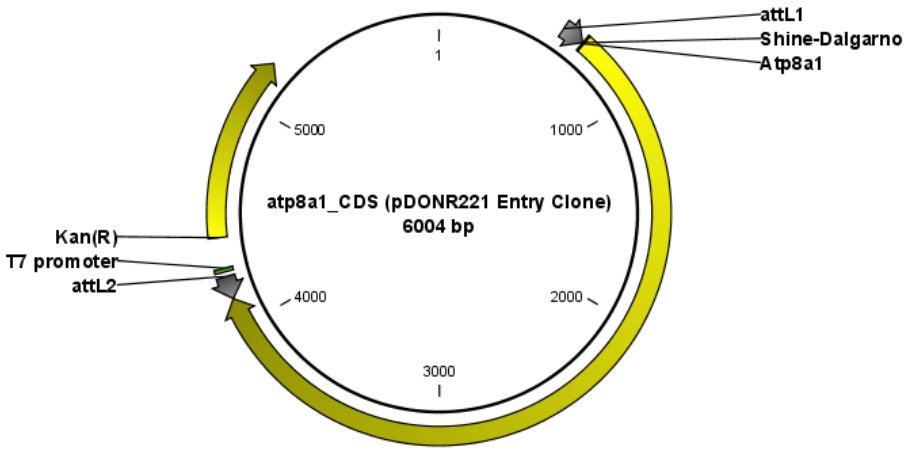


Figure 32.36: *The resulting entry vector opened in a circular view.*

32.4.3 Create expression clones (LR)

The final step in the Gateway cloning work flow is to recombine the entry clone into a destination vector to create an expression clone. Before proceeding to this step, make sure that the sequence of the destination vector was saved in the Navigation Area: find the relevant vector sequence on the Thermo Fisher Scientific's website, copy it, and paste it in in the field that opens when you choose **New | Sequence** in the workbench. Fill in additional information appropriately (enter a "Name", check the "Circular" option) and save the sequence in the Navigation Area.

Note also that for a destination vector to be recognized, it must contain appropriate att sites and the *ccdB* gene. This gene must be present either as a 'ccdB' annotation, or as the exact sequence:

ATGCAGTTTAAGGTTTACACCTATAAAAGAGAGAGCCGTTATCGTCTGTTTGTGGATGTACAGAGTGATATT
ATTGACACGCCCGGGCGACGGATGGTGATCCCCCTGGCCAGTGCACGTCTGCTGTCAGATAAAGTCTCC
CGTGAACTTTACCCGGTGGTGCATATCGGGGATGAAAGCTGGCGCATGATGACCACCGATATGGCCAGT
GTGCCGGTCTCCGTTATCGGGGAAGAAGTGGCTGATCTCAGCCACCGCGAAAATGACATCAAAAACGCC
ATTAACCTGATGTTCTGGGGAATATAA

If the *ccdB* gene is not present or if the sequence is not identical to the above, a solution is to simply add a 'ccdB' annotation. Select part of the vector sequence, right-click and choose 'Add Annotation'. Name the annotation 'ccdB'.

You can now start the tool:

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

In the first step, select one or more entry clones (see how to create an entry clone in section 32.4.2). If you wish to perform separate LR reactions with multiple entry clones, you should run the **Create Expression Clone** in batch mode (see section 8.3).

In the second step, select the destination vector that was previously saved in the Navigation Area (fig 32.37).

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

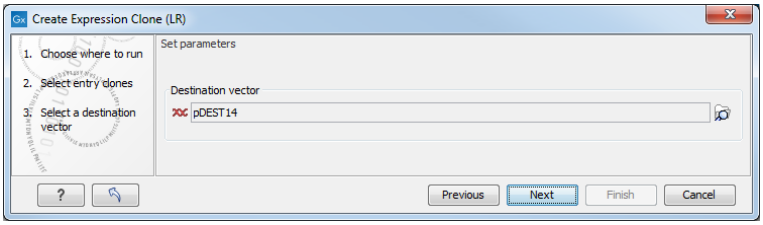


Figure 32.37: *Selecting one or more destination vectors.*

of attL and attR sites is not found, no entry clones will be produced.

When performing multi-site gateway cloning, *Biomedical Genomics Workbench* will insert the fragments (contained in entry clones) by matching the sites that are compatible. If the sites have been defined correctly, an expression clone containing all the fragments will be created. You can find an explanation of the multi-site gateway system at `http://tools.thermofisher.com/downloads/gateway-multisite-seminar.html`

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 32.38.

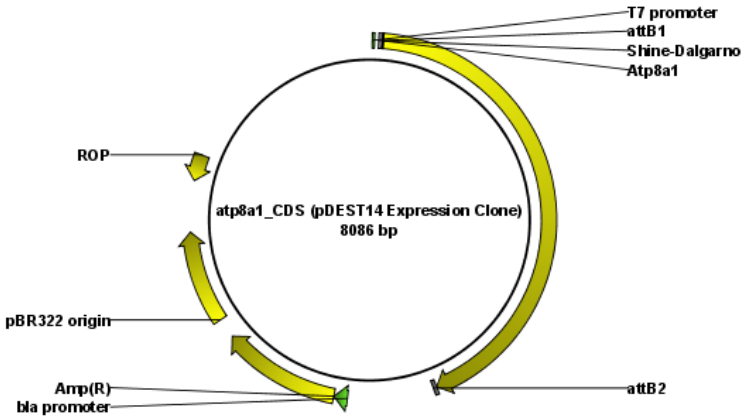


Figure 32.38: *The resulting expression clone opened in a circular view.*

You can choose to create a sequence list with the bi-products as well.

32.5 Gel electrophoresis

Biomedical Genomics 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 32.1.2).
- 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 10.2).
- Ordinary sequence view (see section 10.1).
- Graphical view of sequence lists (see section 10.6).
- Cloning editor (see section 32.3).
- Primer designer (see section 34.3).

- **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 32.39).

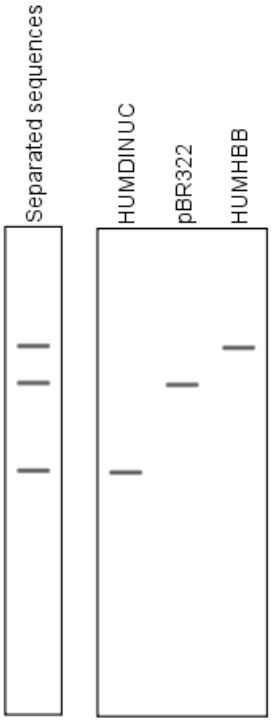


Figure 32.39: *A sequence list shown as a gel.*

32.5.1 Gel view

In figure 32.40 you can see a simulation of a gel with its Side Panel to the right.

Information on bands / fragments You can get information about the individual bands by hovering the mouse cursor on the band of interest. This will display a tool tip with the following information:

- Fragment length
- Fragment region on the original sequence

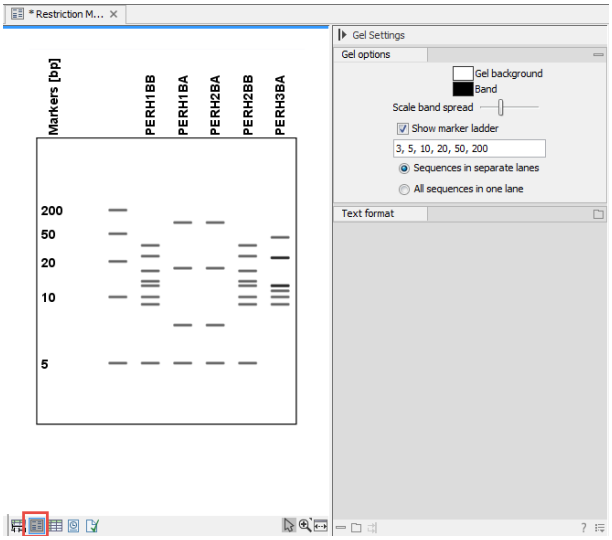


Figure 32.40: *Five lanes showing fragments of five sequences cut with restriction enzymes.*

- 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.

Part X

Sanger Sequencing

Chapter 33

Sequencing Data Analysis

Contents

Biomedical Genomics Workbench lets you import, trim and assemble DNA sequence reads from automated sequencing machines. A number of different formats are supported (see section 6.1). This chapter explains the features in *Biomedical Genomics Workbench* for handling data analysis of low-throughput conventional Sanger sequencing data.

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.

33.1 Importing and viewing trace data

A number of different binary trace data formats can be imported into the program, including *Standard Chromatogram Format (.SCF)*, *ABI sequencer data files (.ABI and .AB1)*, *PHRED output files (.PHD)* and *PHRAP output files (.ACE)* (see section 6.1).

After import, the sequence reads and their trace data are saved as DNA sequences. This means that all analyses that apply to DNA sequences can be performed on the sequence reads.

You can see additional information about the quality of the traces by holding the mouse cursor on the imported sequence. This will display a tool tip as shown in figure 33.1.

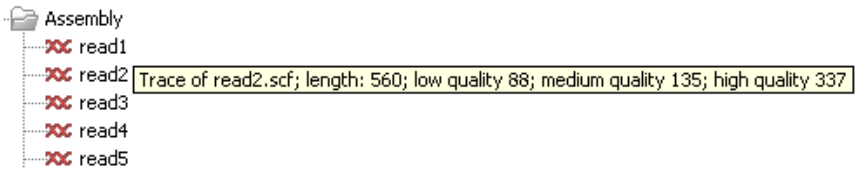


Figure 33.1: *A tooltip displaying information about the quality of the chromatogram.*

The qualities are based on the phred scoring system, with scores below 19 counted as low quality, scores between 20 and 39 counted as medium quality, and those 40 and above counted as high quality.

If the trace file does not contain information about quality, only the sequence length will be shown.

To view the trace data, open the sequence read in a standard sequence view ( ).

The traces can be scaled by dragging the trace vertically as shown in figure figure 33.2. The Workbench automatically adjust the height of the traces to be readable, but if the trace height varies a lot, this manual scaling is very useful.

The height of the area available for showing traces can be adjusted in the **Side Panel** as described insection 33.1.1.

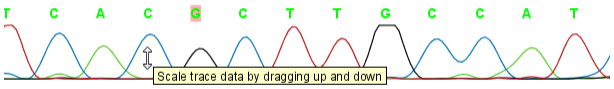


Figure 33.2: *Grab the traces to scale.*

33.1.1 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 33.3):

- **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 33.1.

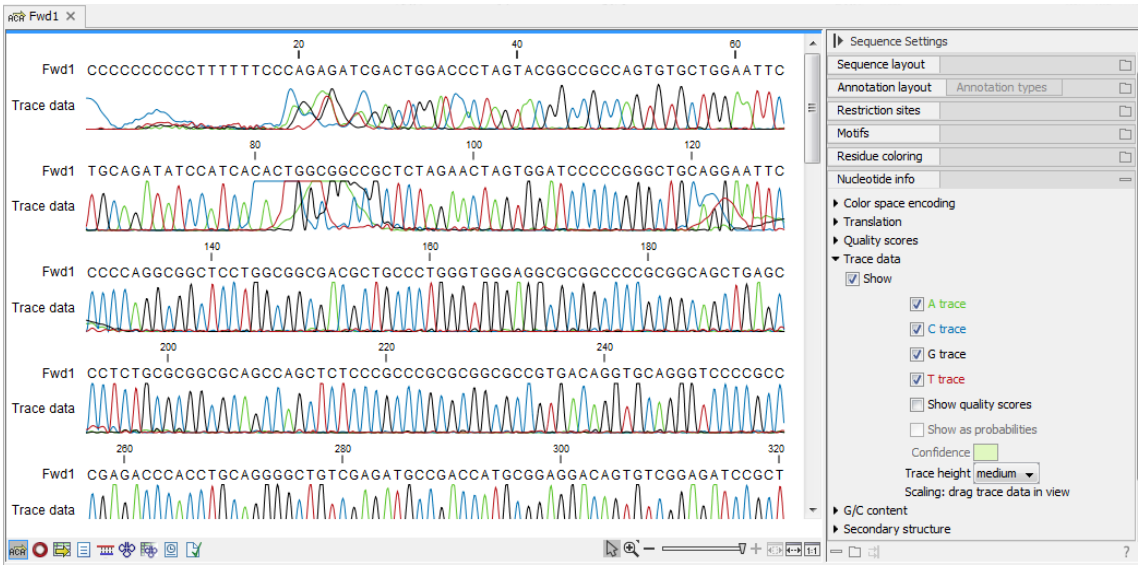


Figure 33.3: *A sequence with trace data. The preferences for viewing the trace are shown in the Side Panel.*

When working with stand alone mappings containing reads with trace data, you can view the traces by turning on the trace setting options as described here **and** choosing **Not compact** in the Read layout setting for the mapping.

Please see section **??**.

33.2 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 33.4). 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 33.3). 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.

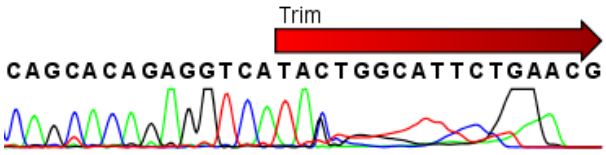


Figure 33.4: *Trimming creates annotations on the regions that will be ignored in the assembly process.*

Note! If you wish to remove regions that are trimmed, you should instead use the NGS trim tool (see section 21.2).

When exporting sequences in fasta format, there is an option to remove the parts of the sequence covered by trim annotations.

33.2.1 Trimming using the Trim tool

Sequence reads can be trimmed based on a number of different criteria. Using a trimming tool for this is particularly useful if:

- You have many sequences to trim.
- You wish to trim vector contamination from sequencing reads.
- You wish to ensure that consistency when trimming. That is, you wish to ensure the same criteria are used for all the sequences in a set.

To start up the Trim tool in the Workbench, go to the menu option:

Toolbox | **Sequencing Data 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 33.5.

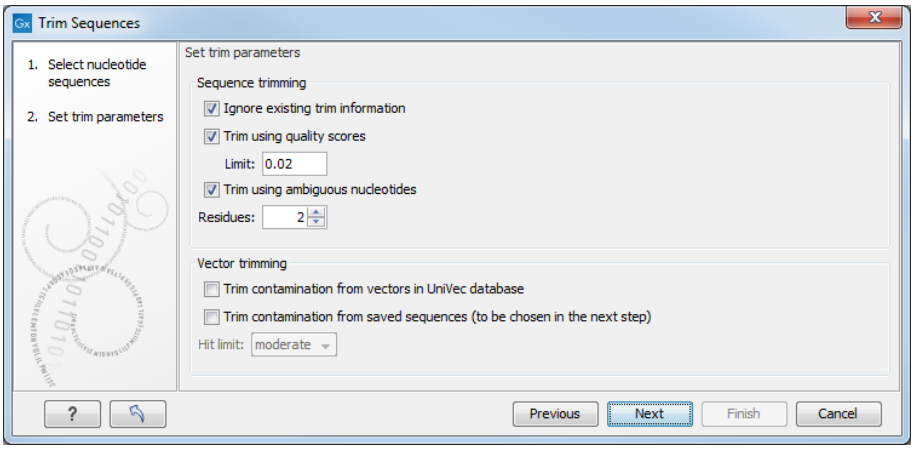


Figure 33.5: *Setting parameters for trimming.*

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: $Q = -10log10(P)$, where P is the base-calling error probability, can then be used to calculate the error probabilities, which in turn can be used to set the limit for, which bases should be trimmed.

  Hence, the first step in the trim process is to convert the quality score (Q) to an error probability: $p_{error} = 10^{\frac{Q}{-10}}$. (This now means that low values are high quality bases.)

  Next, for every base a new value is calculated: $Limit - p_{error}$. This value will be negative for low quality bases, where the error probability is high.

For every base, the Workbench calculates the running sum of this value. If the sum drops below zero, it is set to zero. The part of the sequence not trimmed will be the region ending at the highest value of the running sum and starting at the last zero value before this highest score. Everything before and after this region will be trimmed. A read will be completely removed if the score never makes it above zero.

- **Trim ambiguous nucleotides.** This option trims the sequence ends based on the presence of ambiguous nucleotides (typically N). Note that the automated sequencer generating the data must be set to output ambiguous nucleotides in order for this option to apply. The algorithm takes as input the *maximal number of ambiguous nucleotides allowed in the sequence after trimming*. If this maximum is set to e.g. 3, the algorithm finds the maximum length region containing 3 or fewer ambiguities and then trims away the ends not included in this region. The "Trim ambiguous nucleotides" option trims all types of ambiguous nucleotides (see Appendix G).

- **Trim contamination from vectors in UniVec database.** If selected, the program will match the sequence reads against all vectors in the UniVec database and mark sequence ends with significant matches with a 'Trim' annotation (the database is included when you install the *Biomedical Genomics Workbench*). A list of all the vectors in the UniVec database can be found at http://www.ncbi.nlm.nih.gov/VecScreen/replist.html.

  - **Hit limit.** Specifies how strictly vector contamination is trimmed. Since vector contamination usually occurs at the beginning or end of a sequence, different criteria are applied for terminal and internal matches. A match is considered terminal if it is located within the first 25 bases at either sequence end. Three match categories are defined according to the expected frequency of an alignment with the same score occurring between random sequences. The *Biomedical Genomics Workbench* uses the same settings as VecScreen (http://www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html):
    - **Weak.** Expect 1 random match in 40 queries of length 350 kb
      - Terminal match with Score 16 to 18.
      - Internal match with Score 23 to 24.
    - **Moderate.** Expect 1 random match in 1,000 queries of length 350 kb
      - Terminal match with Score 19 to 23.
      - Internal match with Score 25 to 29.
    - **Strong.** Expect 1 random match in 1,000,000 queries of length 350 kb
      - Terminal match with Score $\geq$ 24.
      - Internal match with Score $\geq$ 30.

    Note that selecting **Weak** will also include matches in the **Moderate** and **Strong** categories.

- **Trim contamination from saved sequences.** This option lets you select your own vector sequences that you have imported into the Workbench. If you select this option, you will be able to select one or more sequences when you click **Next**.

Click **Finish** to start the tool. This will start the trimming process. Views of each trimmed sequence will be shown, and you can inspect the result by looking at the "Trim" annotations (they are colored red as default). Note that the trim annotations are used to signal that this part of the sequence is to be ignored during further analyses, hence the trimmed sequences are not deleted. If there are no trim annotations, the sequence has not been trimmed.

33.2.2 Manual trimming

Sequence reads can be trimmed manually while inspecting their trace and quality data.

> **double-click the sequence to trim in the Navigation Area | select the region you want to trim | right-click the selection | Trim sequence left/right to determine the direction of the trimming**

This will add a trimming annotation to the end of the sequence in the selected direction. No sequence is being deleted here. Rather, the regions covered by trim annotations are noted by downstream analyses (in the same section of the Workbench Toolbox as the Trim tool) as regions to be ignored.

33.3 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 33.4).

Note! You can assemble a maximum of 10000 sequences at a time.

To assemble more sequences, please use the **De Novo Assembly** ( ) tool under **De Novo Sequencing ( )** in the **Toolbox** instead.

To assemble more sequences, you need the *CLC Genomics Workbench* (see http://www.qiagenbioinformatics.com/products/clc-genomics-workbench/).

To perform the assembly: **Toolbox | Sequencing Data 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 33.6

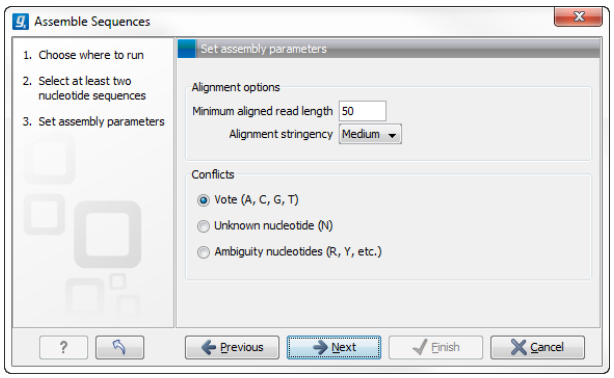


Figure 33.6: *Setting assembly parameters.*

This dialog gives you the following options for assembly:

- **Minimum aligned read length.** The minimum number of nucleotides in a read which must be successfully aligned to the contig. If this criteria is not met by a read, the read is excluded from the assembly.

- **Alignment stringency.** Specifies the stringency 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. Three stringency levels can be set:
  - **Low.**
  - **Medium.**
  - **High.**
- **Conflicts.** If there is a conflict, i.e. a position where there is disagreement about the residue (A, C, T or G), you can specify how the contig sequence should reflect the conflict:
  - **Vote (A, C, G, T).** The conflict will be solved by counting instances of each nucleotide and then letting the majority decide the nucleotide in the contig. In case of equality, ACGT are given priority over one another in the stated order.
  - **Unknown nucleotide (N).** The contig will be assigned an 'N' character in all positions with conflicts (conflicts are registered already when two nucleotides differ).
  - **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the reads (nucleotide ambiguity is registered already when two nucleotides differ). For an overview of ambiguity codes, see Appendix 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 33.7.4.
- **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 33.7.
- **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 33.7.6.
- **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 33.7 on how to use the resulting contigs.

33.4 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 10000 sequences at a time.

To assemble more sequences, please use the **Map Reads to Reference** ( ) under **NGS Core Tools ( )** in the **Toolbox**.

To assemble more sequences, you need the *CLC Genomics Workbench* (see http://www.qiagenbioinformatics.com/products/clc-genomics-workbench/).

To start the assembly:

Toolbox | **Sequencing Data Analysis ( )**| **Assemble Sequences to Reference ( )**

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 33.7

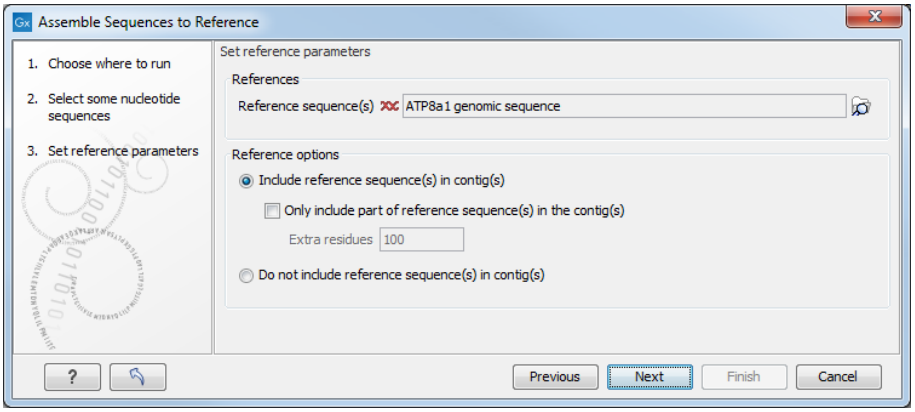


Figure 33.7: *Parameters for how the reference should be handled when assembling sequences to a reference sequence.*

This dialog gives you the following options for assembling:

- **Reference sequence.** Click the **Browse and select element** icon ( ) in order to select one or more sequences to use as reference(s).
- **Include reference sequence(s) in contig(s).** This will create a contig for each reference with the corresponding reference sequence at the top and the aligned sequences below. This option is useful when comparing sequence reads to a closely related reference sequence e.g. when sequencing for SNP characterization.
  - **Only include part of reference sequence(s) in the contig(s).** If the aligned sequences only cover a small part of a reference sequence, it may not be desirable to include the whole reference sequence in a contig. When this option is selected, you can specify

the number of residues from reference sequences that should be included on each side of regions spanned by aligned sequences using the **Extra residues** field.

- **Do not include reference sequence(s) in contig(s).** This will produce contigs without any reference sequence where the input sequences have been assembled using reference sequences as a scaffold. The input sequences are first aligned to the reference sequence(s). Next, the consensus sequence for regions spanned by aligned sequences are extracted and output as contigs. This option is useful when performing assembling sequences where the reference sequences that are not closely related to the input sequencing.

When the reference sequence has been selected, click **Next**, to see the dialog shown in figure 33.8

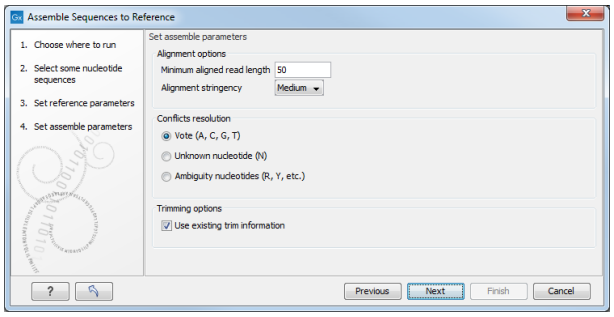


Figure 33.8: *Options for how the input sequences should be aligned and how nucleotide conflicts should be handled.*

In this dialog, you can specify the following options:

- **Minimum aligned read length.** The minimum number of nucleotides in a read which must match a reference sequence. If an input sequence does not meet this criteria, the sequence is excluded from the assembly.

- **Alignment stringency.** Specifies the stringency of the scoring function used for aligning the input sequences to the reference sequence(s). A higher stringency level often produce contigs with lower levels of ambiguity but also reduces the ability to align distant homologs or sequences with a high error rate to reference sequences. The result of a higher stringency level is often that the number of contigs increases and the average length of contigs decreases while the quality of each contig increases. Three stringency levels can be set:

  - **Low.**
  - **Medium.**
  - **High.**

  The stringency settings Low, Medium and High are based on the following score values (mt=match, ti=transition, tv=transversion, un=unknown):

Score values			
	Low	Medium	High
Match (mt)	2	2	2
Transversion (tv)	-6	-10	-20
Transition (ti)	-2	-6	-16
Unknown (un)	-2	-6	-16
Gap	-8	-16	-36

Score Matrix					
	A	C	G	T	N
A	mt	tv	ti	tv	un
C	tv	mt	tv	ti	un
G	ti	tv	mt	tv	un
T	tv	ti	tv	mt	un
N	un	un	un	un	un

- **Conflicts resolution.** If there is a conflict, i.e. a position where aligned sequences disagreement about the residue (A, C, T or G), you can specify how the contig sequence should reflect this conflict:
  - **Unknown nucleotide (N).** The contig will be assigned an 'N' character in all positions with conflicts (conflicts are registered already when two nucleotides differ).
  - **Ambiguity nucleotides (R, Y, etc.).** The contig will display an ambiguity nucleotide reflecting the different nucleotides found in the aligned sequences (nucleotide ambiguity is registered when two nucleotides differ). For an overview of ambiguity codes, see Appendix 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 33.2 for more information about trimming).

Click **Finish** to start the tool. This will start the assembly process. See section 33.7 on how to use the resulting contigs.

33.5 Sort sequences by name

With this functionality you will be able to group sequencing reads based on their file name. A typical example would be that you have a list of files named like this:

```
...
A02__Asp_F_016_2007-01-10
A02__Asp_R_016_2007-01-10
A02__Gln_F_016_2007-01-11
A02__Gln_R_016_2007-01-11
A03__Asp_F_031_2007-01-10
A03__Asp_R_031_2007-01-10
A03__Gln_F_031_2007-01-11
A03__Gln_R_031_2007-01-11
...
```

In this example, the names have five distinct parts (we take the first name as an example):

- **A02** which is the position on the 96-well plate
- **Asp** which is the name of the gene being sequenced
- **F** which describes the orientation of the read (forward/reverse)
- **016** which is an ID identifying the sample
- **2007-01-10** which is the date of the sequencing run

To start mapping these data, you probably want to have them divided into groups instead of having all reads in one folder. If, for example, you wish to map each sample separately, or if you wish to map each gene separately, you cannot simply run the mapping on all the sequences in one step.

That is where **Sort Sequences by Name** comes into play. It will allow you to specify which part of the name should be used to divide the sequences into groups. We will use the example described above to show how it works:

> **Toolbox** | **Molecular Biology Tools** () | **Sequencing Data 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 33.9.

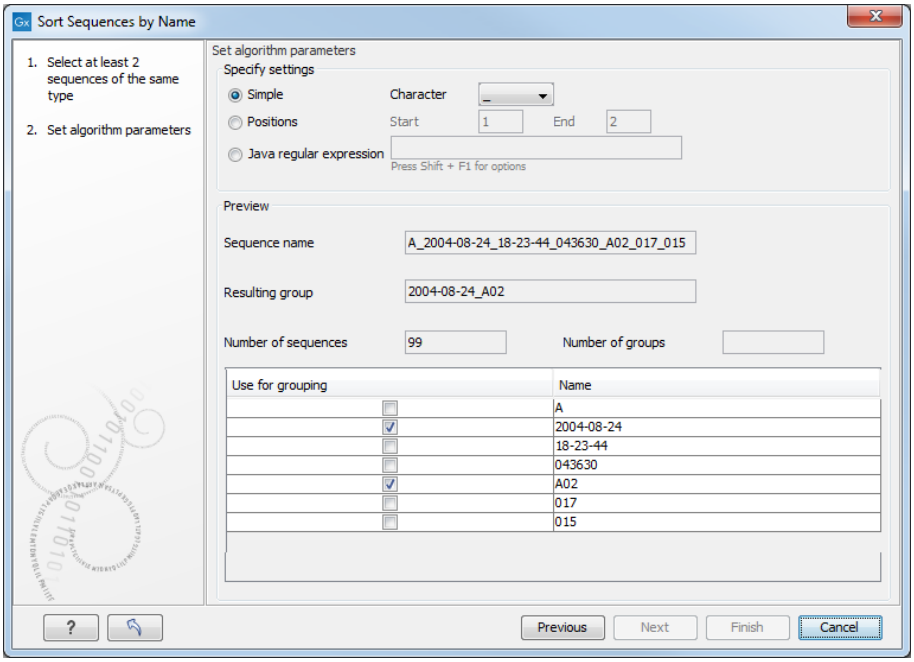


Figure 33.9: *Splitting up the name at every underscore (_) and using the date and analysis position for grouping.*

At the middle of the dialog there is a preview panel listing:

- **Sequence name**. This is the name of the first sequence that has been chosen. It is shown here in the dialog in order to give you a sample of what the names in the list look like.
- **Resulting group**. The name of the group that this sequence would belong to if you proceed with the current settings.
- **Number of sequences**. The number of sequences chosen in the first step.
- **Number of groups**. The number of groups that would be produced when you proceed with the current settings.

This preview cannot be changed. It is shown to guide you when finding the appropriate settings.

Click **Finish** to start the tool. A new sequence list will be generated for each group. It will be named according to the group, e.g. *2004-08-24_A02* will be the name of one of the groups in the example shown in figure 33.9.

Advanced splitting using regular expressions

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 33.9 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 33.10.

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.

33.6 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:

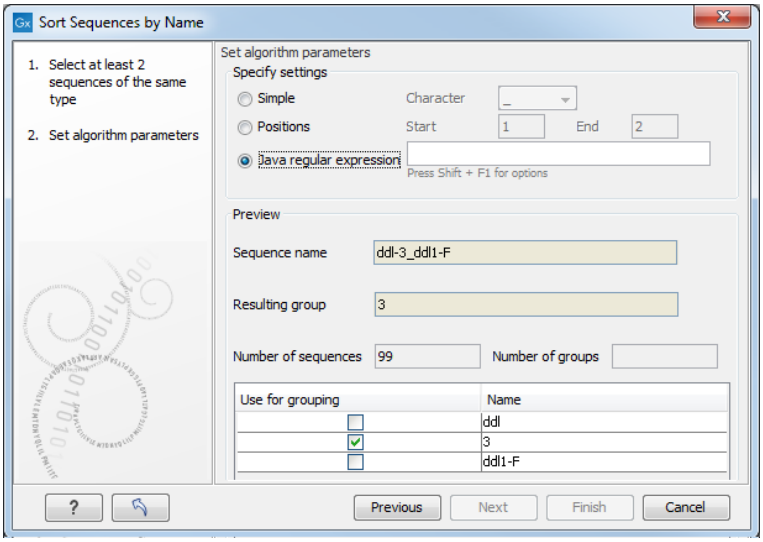


Figure 33.10: *Dividing the sequence into three groups based on the number in the middle of the name.*

Toolbox in the Menu Bar | Sequencing Data Analysis ( )| Add Sequences to Contig ( )

or **right-click in the empty white area of the contig | Add Sequences to Contig ( )**

This opens a dialog where you can 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 33.2.1).

When the elements are selected, click **Next**, and you will see the dialog shown in figure 33.11

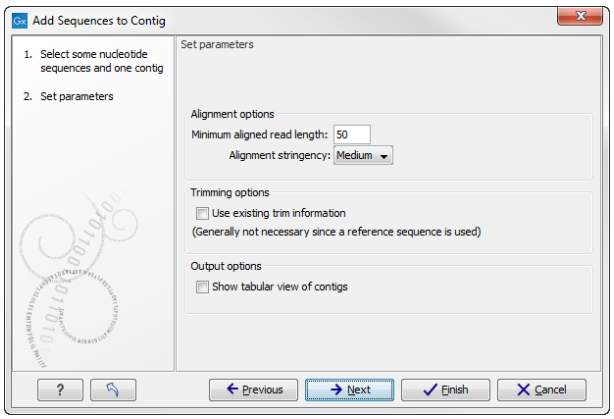


Figure 33.11: *Setting assembly parameters when assembling to an existing contig.*

The options in this dialog are similar to the options that are available when assembling to a reference sequence (see section 33.4).

Click **Finish** to start the tool. This will start the assembly process. See section 33.7 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.

33.7 View and edit contigs and read mappings

The results of the assembly or mapping (assembly to a reference) are respectively contigs or a read mapping. In both cases the sequence reads have been aligned (see figure 33.12). If multiple reference sequences were used, this information will be in a table where the actual visual mapping can be opened by double-clicking.

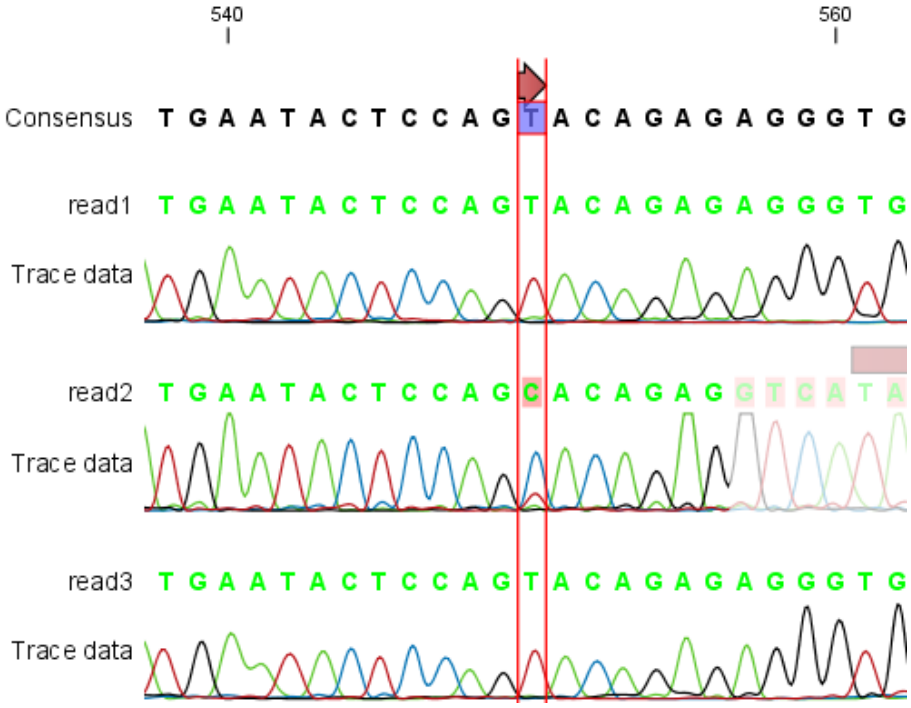


Figure 33.12: *The view of a contig. Notice that you can zoom to a very detailed level.*

You can see that color of the residues and trace at the end of one of the reads has been faded. This indicates, that this region has not contributed to the contig or mapping. This may be due to trimming before or during the assembly or due to misalignment to the other reads.

You can easily adjust the trimmed area to include more of the read in the contig or mapping: simply drag the edge of the faded area as shown in figure 33.13.

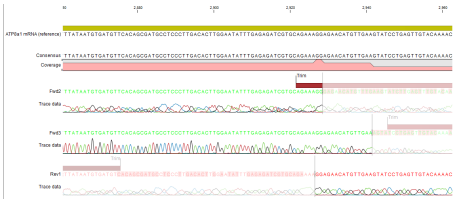


Figure 33.13: *Dragging the edge of the faded area.*

Note! This is only possible when you can see the residues on the reads. This means that you need to have zoomed in to 100% or more and chosen **Compactness** levels "Not compact", "Low"

or "Packed". Otherwise the handles for dragging are not available (this is done in order to make the visual overview more simple).

If reads have been reversed, this is indicated by red. Otherwise, the residues are colored green. The colors can be changed in the **Side Panel** as described in section 33.7.1

If you find out that the reversed reads should have been the forward reads and vice versa, you can reverse complement the whole contig or mapping. Right-click in the empty white area of the contig or mapping and choose to **Reverse Complement Sequence**.

33.7.1 View settings in the Side Panel

The View Settings panel for assemblies and read mappings with fewer than 2000 reads resembles that of alignments (see section **??**) but has some extra preferences described below (figure 33.14).

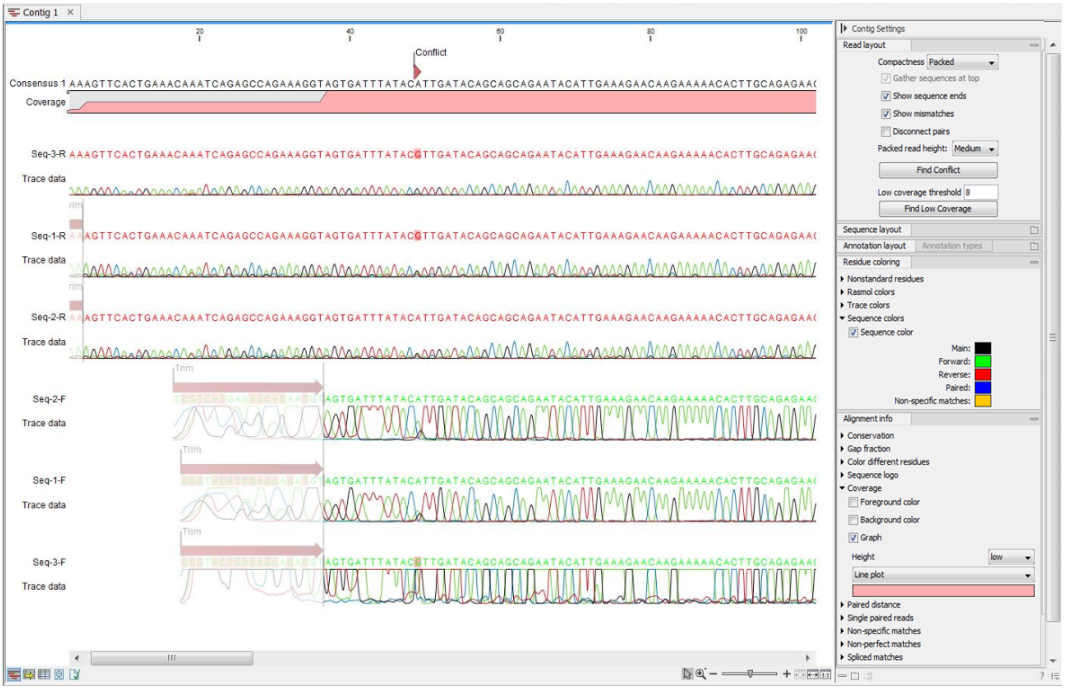


Figure 33.14: *An example of contig, result of an assembly with less than 2000 reads*

- **Read layout.** This section appears at the top of the **Side Panel** when viewing a stand alone read mapping:
  - **Compactness.** The compactness setting options let you control the level of detail to be displayed. This setting affects many of the other settings in the **Side Panel** as well as the general behavior of the view. For example: if the compactness is set to **Compact**, you will not be able to see quality scores or annotations on the reads, even if these are turned on via the "Nucleotide info" palette of the Side Panel. You can change the Compactness setting in the Side Panel directly, or you can use the shortcut: press and hold the Alt key while you scroll with the mouse wheel or touchpad.

  - **Not compact.** This allows the mapping to be viewed in full detail, including quality scores and trace data for the reads, where this is relevant. To view such information, additional viewing options under the **Nucleotide info** view settings must also selected. For further details on these, please see section 33.1.1 and section 10.1.
  - **Low.** Hides trace data, quality scores and puts the reads' annotations on the sequence.
  - **Medium.** The labels of the reads and their annotations are hidden, and the residues of the reads cannot be seen.
  - **Compact.** Even less space between the reads.
  - **Packed.** All the other compactness settings will stack the reads on top of each other, but the packed setting will use all space available for displaying the reads. When zoomed in to 100%, you can see the residues but when zoomed out the reads will be represented as lines just as with the Compact setting. The packed mode is very useful when viewing large amounts of data. However certain functionality possible with other views are not available in packed view. For example, no editing of the read mapping or selections of it can be done and color coding changes are not possible.
- **Gather sequences at top.** Enabling this option affects the view that is shown when scrolling horizontally. If selected, the sequence reads which did not contribute to the visible part of the mapping will be omitted whereas the contributing sequence reads will automatically be placed right below the reference. This setting is not relevant when the compactness is packed.
- **Show sequence ends.** Regions that have been trimmed are shown with faded traces and residues. This illustrates that these regions have been ignored during the assembly.
- **Show mismatches.** When the compactness is packed, you can highlight mismatches which will get a color according to the Rasmol color scheme. A mismatch is whenever the base is different from the reference sequence at this position. This setting also causes the reads that have mismatches to be floated at the top of the view.
- **Disconnect pairs.** This option will break up the paired reads in the display (they are still marked as pairs - this just affects the visualization). The reads are marked with colors for the direction (default red and green) instead of the color for pairs (default blue). This is particularly useful when investigating overlapping pairs in packed view and when the strand / read orientation is important.
- **Packed read height.** When the compactness is set to "packed", you can choose the height of the visible reads. When there are more reads than the height specified, an overflow graph will be displayed below the reads. The overflow graph is shown in the same colors as the sequences, and mismatches in reads are shown as narrow vertical lines. The colors of the small lines represent the mismatching residue. The color codes for the horizontal lines correspond to the color used for highlighting mismatches in the sequences (red = A, blue = C, yellow = G, and green = T), meaning that a red line with half the height of the blue part of the overflow graph will represent a mismatching "A" in half of the paired reads at this particular position.
- **Find Conflict.** Clicking this button selects the next position where there is an conflict between the sequence reads. Residues that are different from the reference are colored (as default), providing an overview of the conflicts. Since the nextF conflict is

automatically selected it is easy to make changes. You can also use the Space key to find the next conflict.

- **Low coverage threshold.** All regions with coverage up to and including this value are considered low coverage. When clicking the 'Find low coverage' button the next region in the read mapping with low coverage will be selected.

- **Alignment info.** There is one additional parameter:
  - **Coverage**: Shows how many sequence reads that are contributing information to a given position in the mapping. The level of coverage is relative to the overall number of sequence reads.
    - **Foreground color.** Colors the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage.
    - **Background color.** Colors the background of the letters using a gradient, where the left side color is used for low coverage and the right side is used for maximum coverage
    - **Graph.** The coverage is displayed as a graph (Learn how to export the data behind the graph in section 6.8).
      - **Height.** Specifies the height of the graph.
      - **Type.** The graph can be displayed as Line plot, Bar plot or as a Color bar.
      - **Color box.** For Line and Bar plots, the color of the plot can be set by clicking the color box. If a Color bar is chosen, the color box is replaced by a gradient color box as described under Foreground color.

- **Residue coloring.** There is one additional parameter:
  - **Sequence colors.** This option lets you use different colors for the reads.
    - **Main**. The color of the consensus and reference sequence. Black per default.
    - **Forward**. The color of forward reads (single reads). Green per default.
    - **Reverse**. The color of reverse reads (single reads). Red per default.
    - **Paired**. The color of paired reads. Blue per default. Note that reads from **broken pairs** are colored according to their Forward/Reverse orientation or as a Non-specific match, but with a darker nuance than ordinary single reads.
    - **Non-specific matches**. When a read would have matched equally well another place in the mapping, it is considered a non-specific match. This color will "overrule" the other colors. Note that if you are mapping with several reference sequences, a read is considered a double match when it matches more than once *across all the contigs/references*. A non-specific match is yellow per default.

- **Sequence layout.**
  - **Matching residues as dots** Matching residues will be presented as dots. Only the top sequence will be preserved in its original format.

There are many other viewing options available, both general and aimed as specific elements of a contig or a mapping, which can be adjusted in the View settings. Those covered here were the key ones relevant for standard review of the results.

33.7.2 Editing a contig or read mapping

When editing contigs and read mappings, you are typically interested in confirming or changing single bases, and this can be done simply by:

selecting the base | **typing the right base**

Some users prefer to use lower-case letters in order to be able to see which bases were altered when they use the results later on. In *Biomedical Genomics Workbench* all changes are recorded in the history log (see section **??**) allowing the user to quickly reconstruct the actions performed in the editing session.

There are three shortcut keys for easily finding the positions where there are conflicts:

- Space bar: Finds the *next* conflict.
- "." (punctuation mark key): Finds the *next* conflict.
- "," (comma key): Finds the *previous* conflict.

In the contig or mapping view, you can use **Zoom in** () to zoom to a greater level of detail than in other views (see figure 33.12). This is useful for discerning the trace curves.

If you want to replace a residue with a gap, use the **Delete** key.

If you wish to edit a selection of more than one residue:

right-click the selection | **Edit Selection (**  **)**

This will show a warning dialog, but you can choose never to see this dialog again by clicking the checkbox at the bottom of the dialog.

Note that for contigs or mappings with more than 1000 reads, you can only do single-residue replacements (you can't delete or edit a selection). When the compactness is **Packed**, you cannot edit any of the reads.

33.7.3 Sorting reads

If you wish to change the order of the sequence reads, simply drag the label of the sequence up and down. Note that this is not possible if you have chosen **Gather sequences at top** or set the compactness to **Packed** in the **Side Panel**.

You can also sort the reads by right-clicking a sequence label and choose from the following options:

- **Sort Reads by Alignment Start Position.** This will list the first read in the alignment at the top etc.
- **Sort Reads by Name.** Sort the reads alphabetically.
- **Sort Reads by Length.** The shortest reads will be listed at the top.

33.7.4 Read conflicts

After assembly or mapping, conflicts between the reads are annotated on the consensus sequence. The definition of a conflict is *a position where at least one of the reads has a different residue compared to the reference*.

A conflict can be in two states:

- **Conflict**. Both the annotation and the corresponding row in the Table ( ) are colored **red**.
- **Resolved**. Both the annotation and the corresponding row in the Table ( ) are colored **green**.

The conflict can be resolved by correcting the deviating residues in the reads as described above.

A fast way of making all the reads reflect the consensus sequence is to select the position in the consensus, right-click the selection, and choose **Transfer Selection to All Reads**.

The opposite is also possible: make a selection on one of the reads, right click, and **Transfer Selection to Contig Sequence**.

33.7.5 Extract parts of a mapping

Sometimes it is useful to extract part of a mapping for in-depth analysis. This could be the case if you have performed an analysis of a whole genome data set and have found a region that you are particularly interested in analyzing further. Rather than running all further analysis on your full data, you may prefer to run only on a subset of the data. You can extract a subset of your mapping data by running the **Extract from Selection** tool on a selected region in your mapping. The result of running this tool is a new mapping which contains only the reads (and optionally only those that are of a particular type) in your selected region.

To select a region, use the **Selection mode** ( ) (see Section 2.2.3 for a detailed description of the different modes) and select you region of interest in your mapping, then right-click. You are now presented with the dialog shown in figure 33.15.

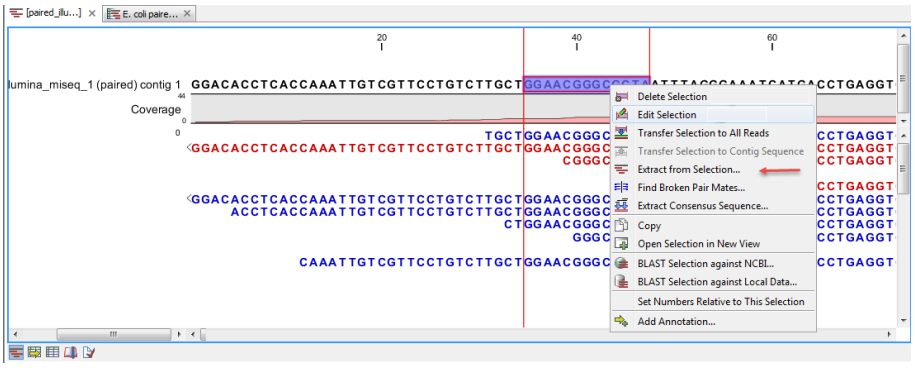


Figure 33.15: *Extracting parts of a mapping.*

When you choose the **Extract from Selection** option you are presented by the dialog shown in figure 33.16.

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:

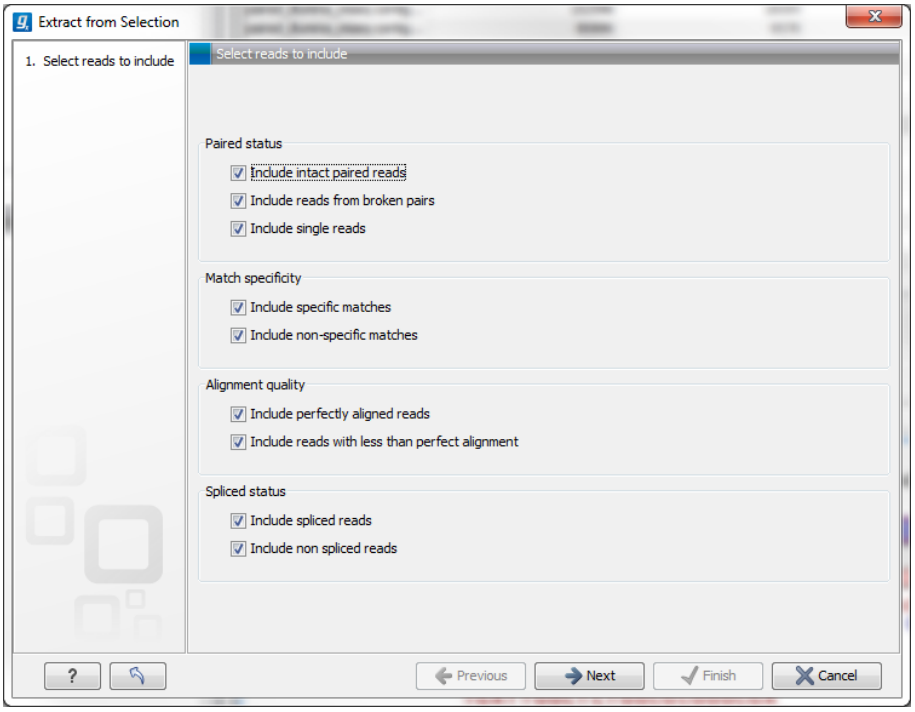


Figure 33.16: *Selecting the reads to include.*

Paired status **Include intact paired reads** When paired reads are placed within the paired distance specified, they will fall into this category. Per default, these reads are colored in blue.

Include paired reads from broken pairs When a pair is broken, either because only one read in the pair matches, or because the distance or relative orientation is wrong, the reads are placed and colored as single reads, but you can still extract them by checking this box.

Include single reads This will include reads that are marked as single reads (as opposed to paired reads). Note that paired reads that have been broken during assembly are not included in this category. Single reads that come from trimming paired sequence lists are included in this category.

Match specificity **Include specific matches** Reads that only are mapped to one position.

Include non-specific matches Reads that have multiple equally good alignments to the reference. These reads are colored yellow per default.

Alignment quality **Include perfectly aligned reads** Reads where *the full read* is perfectly aligned to the reference sequence (or consensus sequence for de novo assemblies). Note that at the end of the contig, reads may extend beyond the contig (this is not visible unless you make a selection on the read and observe the position numbering in the status bar). Such reads are not considered perfectly aligned reads because they don't align in their entire length.

Include reads with less than perfect alignment Reads with mismatches, insertions or deletions, or with unaligned nucleotides at the ends (the faded part of a read).

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 31.1)

You will now have all paired reads from the original mapping in a list.

33.7.6 Variance table

In addition to the standard graphical display of a contig or mapping as described above, you can also see a tabular overview of the conflicts between the reads by clicking the **Table (** **)** icon at the bottom of the view.

This will display a new view of the conflicts as shown in figure 33.17.

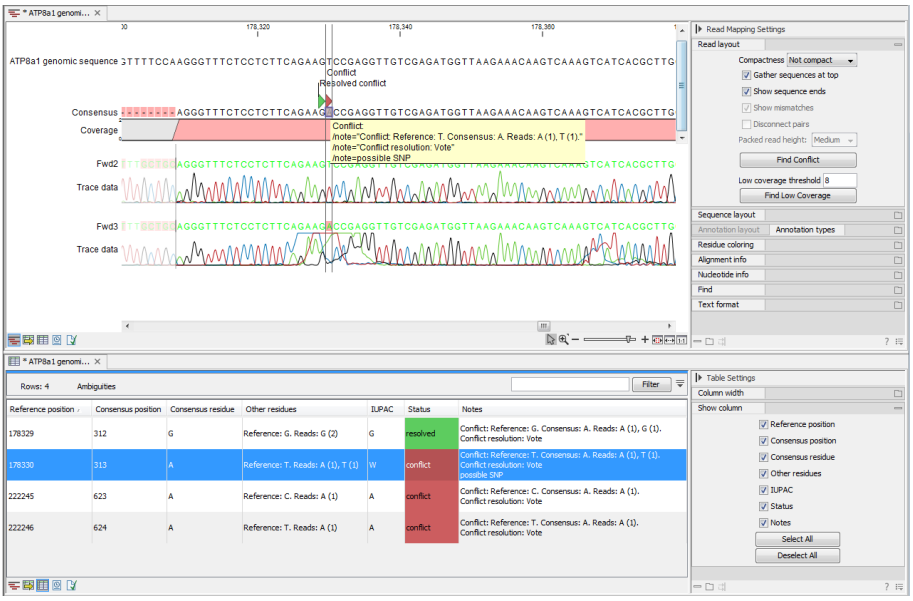


Figure 33.17: *The graphical view is displayed at the top, and underneath the conflicts are shown in a table. At the conflict at position 313, the user has entered a comment in the table (to see it, make sure the Notes column is wide enough to display all text lines). This comment is now also added to the tooltip of the conflict annotation in the graphical view above.*

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 33.17, you can see that at position 637 there is a 'C' in the top read in the graphical view. The other two reads have a 'T'. Therefore, the table displays the following text: 'C (1), T (2)'.
- **IUPAC.** The ambiguity code for this position. The ambiguity code reflects the residues in the reads - not in the consensus sequence. (The IUPAC codes can be found in section G.)
- **Status.** The status can either be conflict or resolved:
  - **Conflict.** Initially, all the rows in the table have this status. This means that there is one or more differences between the sequences at this position.
  - **Resolved.** If you edit the sequences, e.g. if there was an error in one of the sequences, and they now all have the same residue at this position, the status is set to *Resolved*.
- **Note.** Can be used for your own comments on this conflict. Right-click in this cell of the table to add or edit the comments. The comments in the table are associated with the conflict annotation in the graphical view. Therefore, the comments you enter in the table will also be attached to the annotation on the consensus sequence (the comments can be displayed by placing the mouse cursor on the annotation for one second - see figure 33.17). 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.

33.8 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 | Sequencing Data Analysis ( )| Reassemble Contig ( ) | select the contig from Navigation Area, move to 'Selected Elements' and click Next

or **right-click in the empty white area of the contig | Reassemble contig ( )**

This opens a dialog as shown in figure 33.18

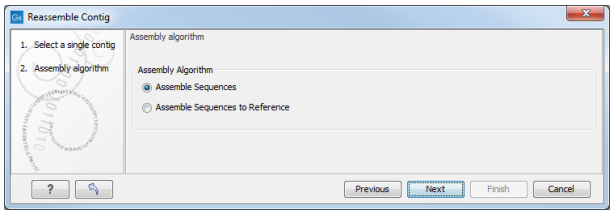


Figure 33.18: *Re-assembling a contig.*

In this dialog, you can choose:

- **De novo assembly**. This will perform a normal assembly in the same way as if you had selected the reads as individual sequences. When you click **Next**, you will follow the same steps as described in section 33.3. 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 33.4.

When you click **Finish**, a new contig is created, so you do not lose the information in the old contig.

33.9 Secondary peak calling

Biomedical Genomics Workbench is able to detect secondary peaks - a peak within a peak - to help discover heterozygous mutations. Looking at the height of the peak below the top peak, the *Biomedical Genomics Workbench* considers all positions in a sequence, and if a peak is higher than the threshold set by the user, it will be "called".

The peak 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 33.2).

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 | **Sequencing Data 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 33.19.

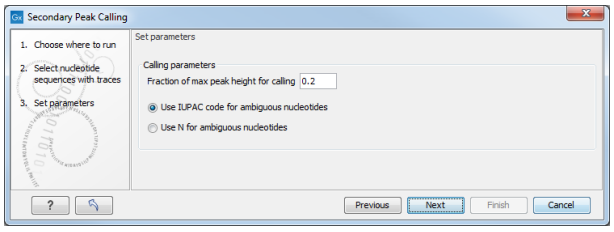


Figure 33.19: *Setting parameters secondary peak calling.*

The following parameters can be adjusted in the dialog:

- **Fraction of max peak height for calling.** Adjust this value to specify how high the secondary peak must be to be called.
- **Use IUPAC code / N for ambiguous nucleotides.** When a secondary peak is called, the residue at this position can either be replaced by an N or by a ambiguity character based on the IUPAC codes (see section G).

Clicking **Next** allows you to add annotations. In addition to changing the actual sequence, annotations can be added for each base that has been called. The annotations hold information about the fraction of the max peak height.

Click **Finish** to start the tool. This will start the secondary peak calling. A detailed history entry will be added to the history specifying all the changes made to the sequence.

Chapter 34

Primers

Contents

Biomedical Genomics Workbench offers graphically and algorithmically advanced design of primers and probes for various purposes. This chapter begins with a brief introduction to the general concepts of the primer designing process. Then follows instructions on how to adjust parameters for primers, how to inspect and interpret primer properties graphically and how to interpret, save

and analyze the output of the primer design analysis. After a description of the different reaction types for which primers can be designed, the chapter closes with sections on how to match primers with other sequences and how to create a primer order.

34.1 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 34.1), the basic options for viewing the template sequence are the same as for the standard sequence view (see section 10.1 for an explanation of these options). 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 34.1: *The initial view of the sequence used for primer design.*

34.1.1 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 34.2).

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 fulfils all criteria and a red circle indicates a primer which fails to meet one or more of the set criteria. For more detailed information, place the mouse cursor over the circle representing the primer of interest. A tool-tip will then appear on screen, displaying detailed information about the primer in relation to the set criteria. To locate the primer on the sequence,

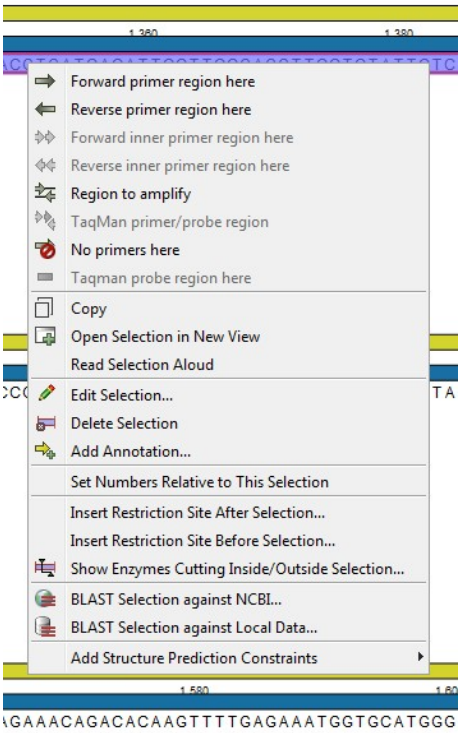


Figure 34.2: *Right-click menu allowing you to specify regions for the primer design*

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.

34.1.2 Scoring primers

Biomedical Genomics Workbench employs a proprietary algorithm to rank primer and probe solutions. The algorithm considers both the parameters pertaining to single oligos, such as e.g. the secondary structure score and parameters pertaining to oligo-pairs such as e.g. the oligo pair-annealing score. The ideal score for a solution is 100 and solutions are thus ranked in descending order. Each parameter is assigned an ideal value and a tolerance. Consider for example oligo self-annealing, here the ideal value of the annealing score is 0 and the tolerance corresponds to the maximum value specified in the side panel. The contribution to the final score is determined by how much the parameter deviates from the ideal value and is scaled by the specified tolerance. Hence, a large deviation from the ideal and a small tolerance will give a large deduction in the final score and a small deviation from the ideal and a high tolerance will give a small deduction in the final score.

34.2 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 34.3).

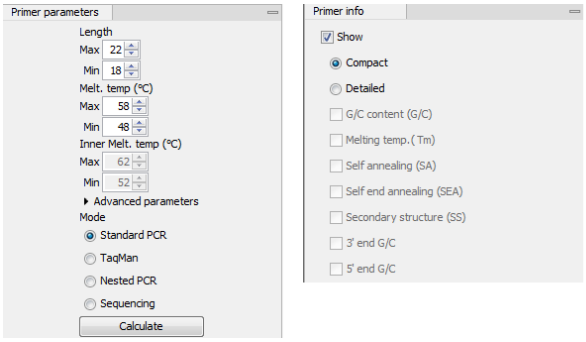


Figure 34.3: *The two groups of primer parameters (in the program, the Primer information group is listed below the other group).*

34.2.1 Primer Parameters

In this preference group a number of criteria can be set, which the selected primers must meet. All the criteria concern *single primers*, as primer pairs are not generated until the **Calculate** button is pressed. Parameters regarding primer and probe sets are described in detail for each reaction mode (see below).

- **Length.** Determines the length interval within which primers can be designed by setting a maximum and a minimum length. The upper and lower lengths allowed by the program are 50 and 10 nucleotides respectively.
- **Melting temperature.** Determines the temperature interval within which primers must lie. When the *Nested PCR* or *TaqMan* reaction type is chosen, the first pair of melting temperature interval settings relate to the outer primer pair i.e. not the probe. Melting temperatures are calculated by a nearest-neighbor model which considers stacking interactions between

neighboring bases in the primer-template complex. The model uses state-of-the-art thermodynamic parameters [SantaLucia, 1998] and considers the important contribution from the dangling ends that are present when a short primer anneals to a template sequence [Bommarito et al., 2000]. A number of parameters can be adjusted concerning the reaction mixture and which influence melting temperatures (see below). Melting temperatures are corrected for the presence of monovalent cations using the model of [SantaLucia, 1998] and temperatures are further corrected for the presence of magnesium, deoxynucleotide triphosphates (dNTP) and dimethyl sulfoxide (DMSO) using the model of [von Ahsen et al., 2001].

- **Inner melting temperature.** This option is only activated when the *Nested PCR* or *TaqMan* mode is selected. In *Nested PCR* mode, it determines the allowed melting temperature interval for the inner/nested pair of primers, and in *TaqMan* mode it determines the allowed temperature interval for the TaqMan probe.

- **Advanced parameters.** A number of less commonly used options

  - **Buffer properties.** A number of parameters concerning the reaction mixture which influence melting temperatures.

    * **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM). Note that in the case of a mix of primers, the concentration here refers to the individual primer and not the combined primers concentration.
    * **Salt concentration.** Specifies the concentration of monovalent cations ($[NA^+]$, $[K^+]$ and equivalents) in units of millimoles (mM)
    * **Magnesium concentration.** Specifies the concentration of magnesium cations ($[Mg^{++}]$) in units of millimoles (mM)
    * **dNTP concentration.** Specifies the combined concentration of all deoxynucleotide triphosphates in units of millimoles (mM)
    * **DMSO concentration.** Specifies the concentration of dimethyl sulfoxide in units of volume percent ($vol.\%$)

  - **GC content.** Determines the interval of CG content (% C and G nucleotides in the primer) within which primers must lie by setting a maximum and a minimum GC content.

  - **Self annealing.** Determines the maximum self annealing value of all primers and probes. This determines the amount of base-pairing allowed between two copies of the same molecule. The self annealing score is measured in number of hydrogen bonds between two copies of primer molecules, with A-T base pairs contributing 2 hydrogen bonds and G-C base pairs contributing 3 hydrogen bonds.

  - **Self end annealing.** Determines the maximum self end annealing value of all primers and probes. This determines the number of consecutive base pairs allowed between the 3' end of one primer and another copy of that primer. This score is calculated in number of hydrogen bonds (the example below has a score of 4 - derived from 2 A-T base pairs each with 2 hydrogen bonds).

```
AATTCCCTACAATCCCCAAA
                  ||
  AAACCCCTAACATCCCTTAA
```

.

  - **Secondary structure.** Determines the maximum score of the optimal secondary DNA structure found for a primer or probe. Secondary structures are scored by the number of hydrogen bonds in the structure, and 2 extra hydrogen bonds are added for each stacking base-pair in the structure.

- **3' end G/C restrictions.** When this checkbox is selected it is possible to specify restrictions concerning the number of G and C molecules in the 3' end of primers and probes. A low G/C content of the primer/probe 3' end increases the specificity of the reaction. A high G/C content facilitates a tight binding of the oligo to the template but also increases the possibility of mispriming. Unfolding the preference groups yields the following options:

  - **End length.** The number of consecutive terminal nucleotides for which to consider the C/G content
  - **Max no. of G/C.** The maximum number of G and C nucleotides allowed within the specified length interval
  - **Min no. of G/C.** The minimum number of G and C nucleotides required within the specified length interval

- **5' end G/C restrictions.** When this checkbox is selected it is possible to specify restrictions concerning the number of G and C molecules in the 5' end of primers and probes. A high G/C content facilitates a tight binding of the oligo to the template but also increases the possibility of mis-priming. Unfolding the preference groups yields the same options as described above for the 3' end.

- **Mode.** Specifies the reaction type for which primers are designed:

  - **Standard PCR.** Used when the objective is to design primers, or primer pairs, for PCR amplification of a single DNA fragment.
  - **Nested PCR.** Used when the objective is to design two primer pairs for nested PCR amplification of a single DNA fragment.
  - **Sequencing.** Used when the objective is to design primers for DNA sequencing.
  - **TaqMan.** Used when the objective is to design a primer pair and a probe for TaqMan quantitative PCR.

  Each mode is described further below.

- **Calculate.** Pushing this button will activate the algorithm for designing primers

34.3 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 34.3).

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.

34.3.1 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 34.4).

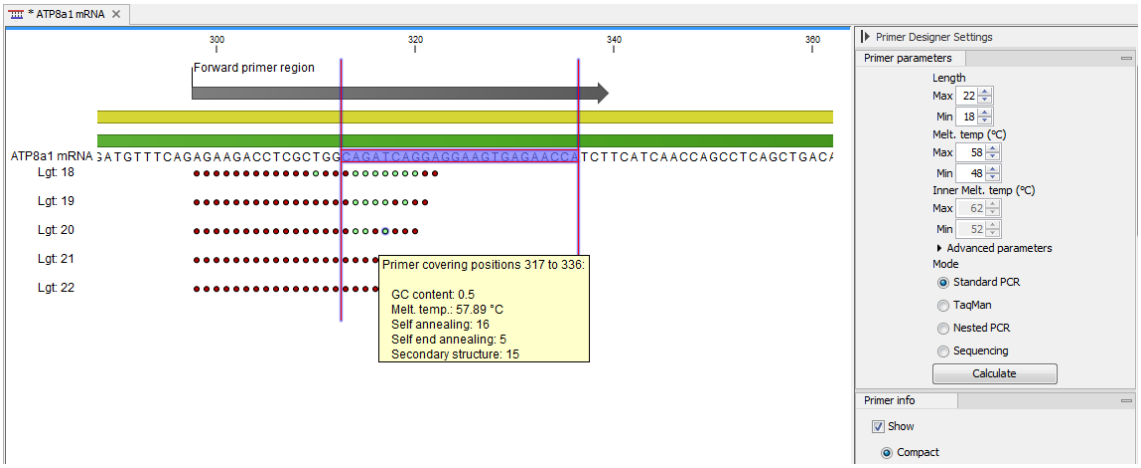


Figure 34.4: *Compact information mode.*

The number of information lines reflects the chosen length interval for primers and probes. One line is shown for every possible primer-length, if the length interval is widened more lines will appear. At each potential primer starting position a circle is shown which indicates whether the primer fulfills the requirements set in the primer parameters preference group. A green primer indicates a primer which fulfils all criteria and a red primer indicates a primer which fails to meet one or more of the set criteria. For more detailed information, place the mouse cursor over the circle representing the primer of interest. A tool-tip will then appear on screen displaying detailed information about the primer in relation to the set criteria. To locate the primer on the sequence, simply left-click the circle using the mouse.

The various primer parameters can now be varied to explore their effect and the view area will dynamically update to reflect this. If e.g. the allowed melting temperature interval is widened more green circles will appear indicating that more primers now fulfill the set requirements and if e.g. a requirement for 3' G/C content is selected, rec circles will appear at the starting points of the primers which fail to meet this requirement.

34.3.2 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 34.5).

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

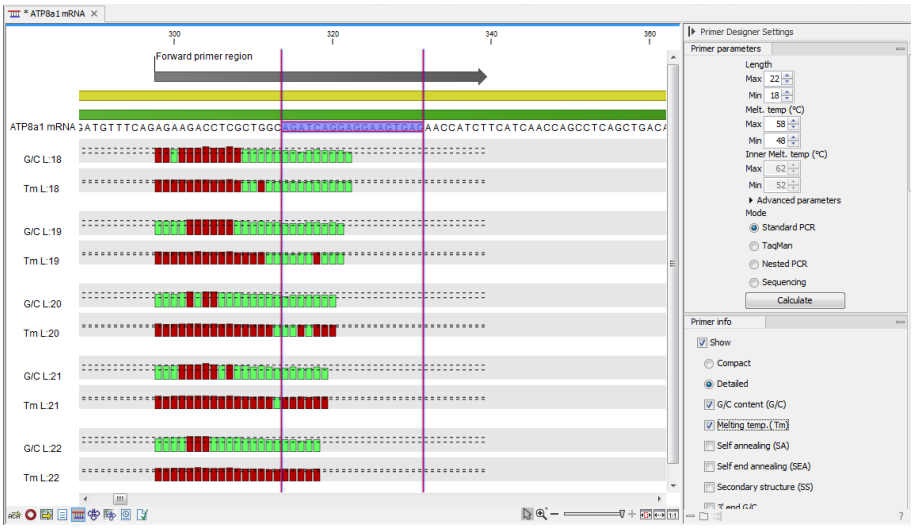


Figure 34.5: *Detailed information mode.*

- 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.

34.4 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 34.6).

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

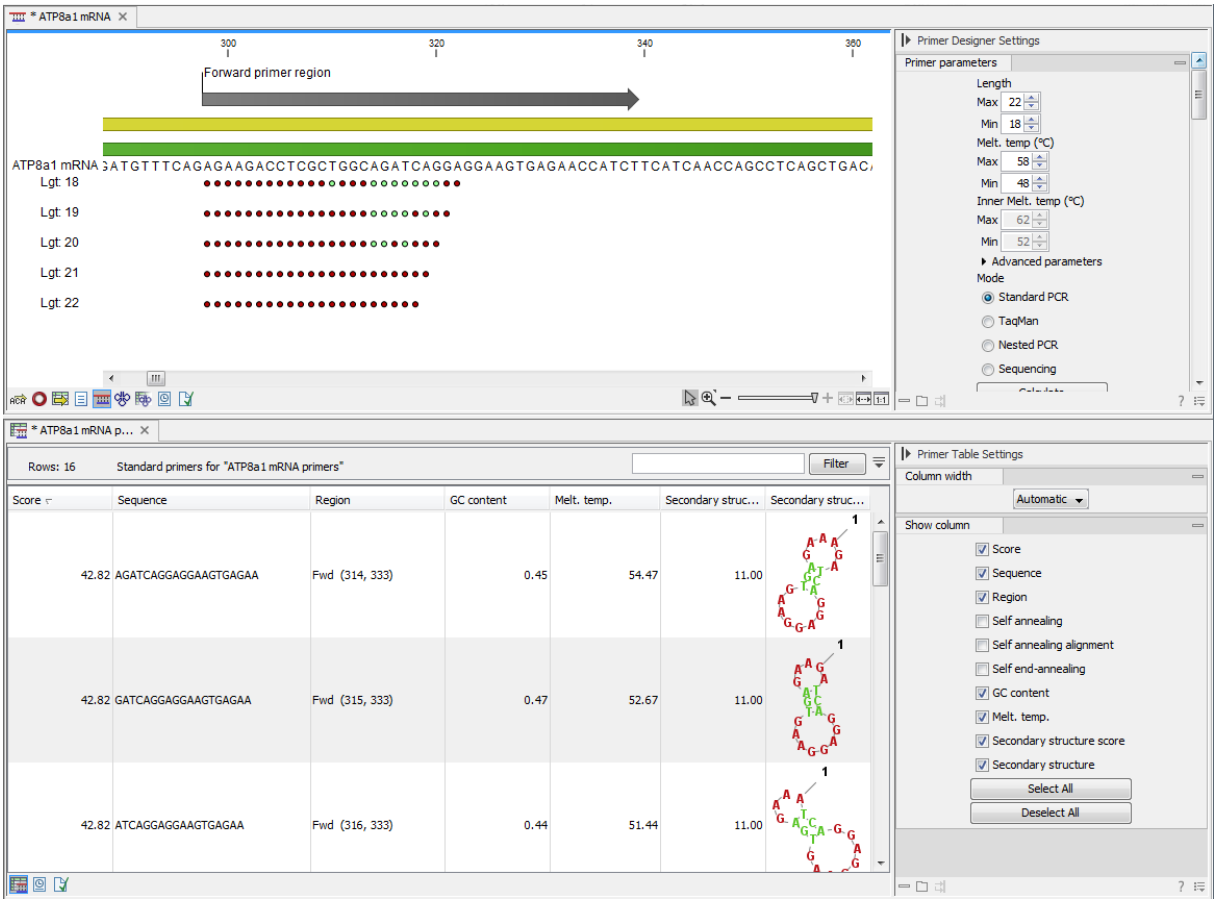


Figure 34.6: *Proposed primers.*

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**.

34.5 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 34.3) and setting the desired parameter values in the Primer Parameters preference group, the **Calculate** button will activate the primer design algorithm.

34.5.1 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 34.7).

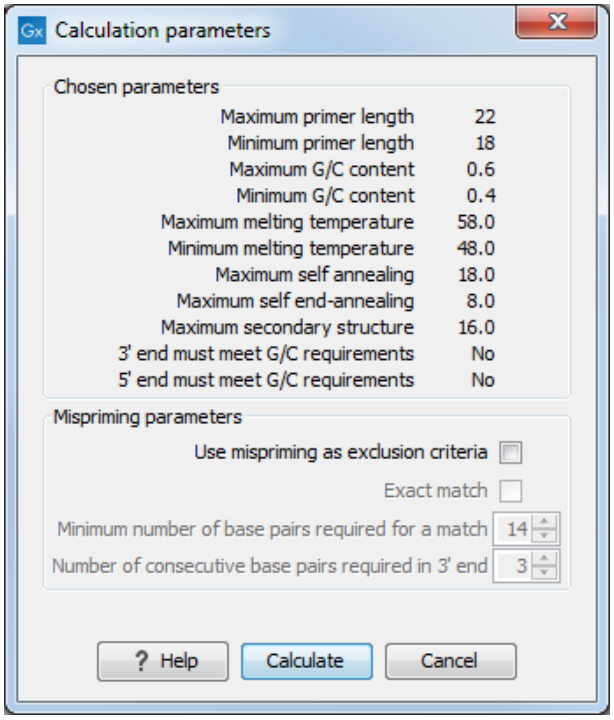


Figure 34.7: *Calculation dialog for PCR primers when only a single primer region has been defined.*

The top part of this dialog shows the parameter settings chosen in the Primer parameters preference group which will be used by the design algorithm.

Mispriming: The lower part contains a menu where the user can choose to include mispriming as an exclusion criteria in the design process. If this option is selected the algorithm will search for competing binding sites of the primer within the rest of the sequence, to see if the primer would match to multiple locations. If a competing site is found (according to the parameters set), the primer will be excluded.

The adjustable parameters for the search are:

- **Exact match**. Choose only to consider exact matches of the primer, i.e. all positions must base pair with the template for mispriming to occur.

- **Minimum number of base pairs required for a match**. How many nucleotides of the primer that must base pair to the sequence in order to cause mispriming.
- **Number of consecutive base pairs required in 3' end**. How many consecutive 3' end base pairs in the primer that MUST be present for mispriming to occur. This option is included since 3' terminal base pairs are known to be essential for priming to occur.

Note! Including a search for potential mispriming sites will prolong the search time substantially if long sequences are used as template and if the minimum number of base pairs required for a match is low. If the region to be amplified is part of a very long molecule and mispriming is a concern, consider extracting part of the sequence prior to designing primers.

34.5.2 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 34.8).

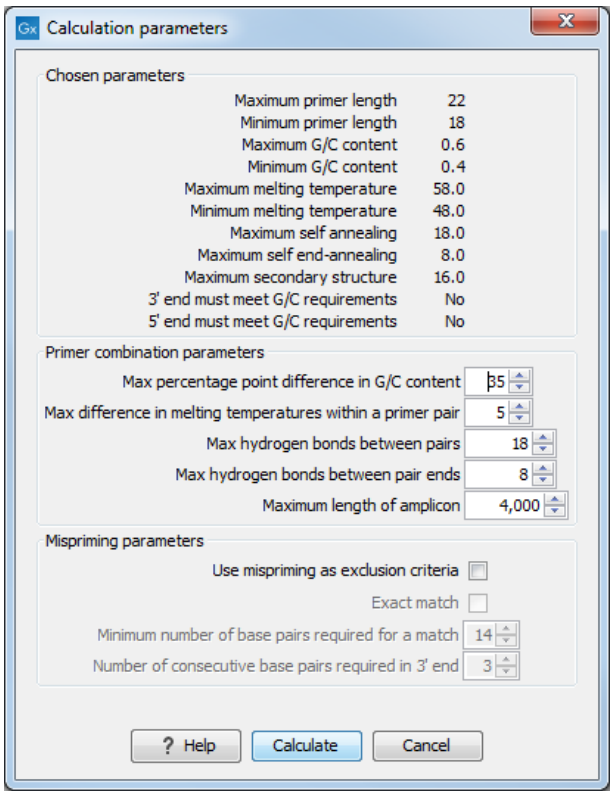


Figure 34.8: *Calculation dialog for PCR primers when two primer regions have been defined.*

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 34.5.1). 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.

34.5.3 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

34.6 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 34.3) 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 34.9).

The top and bottom parts of this dialog are identical to the *Standard PCR* dialog for designing primer pairs described above.

The central part of the dialog contains parameters pertaining to primer pairs and the comparison between the outer and the inner pair. Here five options can be set:

- Maximum percentage point difference in G/C content (described above under Standard PCR) - this criteria is applied to both primer pairs independently.
- Maximal difference in melting temperature of primers in a pair - the number of degrees Celsius that primers in a pair are all allowed to differ. This criteria is applied to both primer

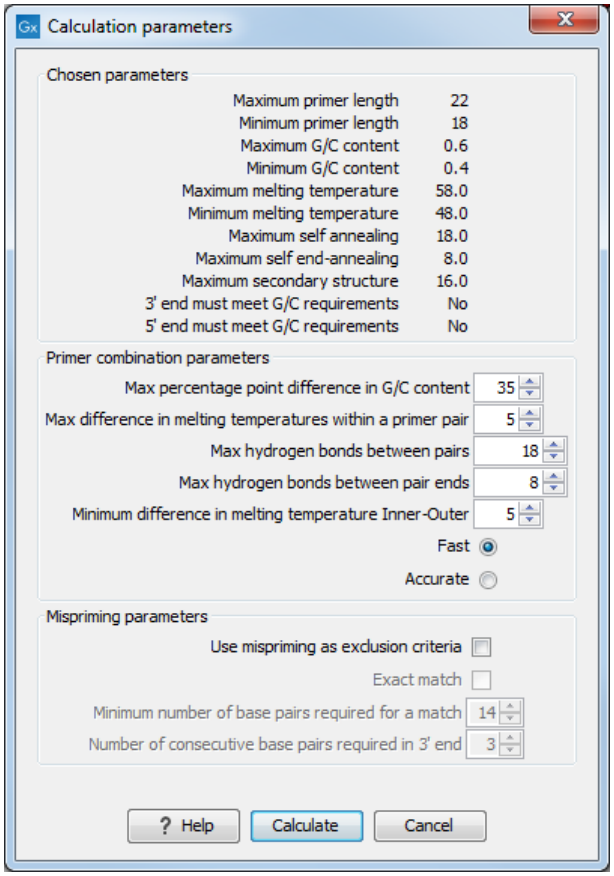


Figure 34.9: *Calculation dialog for nested primers.*

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.

34.7 TaqMan

Biomedical Genomics Workbench allows the user to design primers and probes for TaqMan PCR applications.

TaqMan probes are oligonucleotides that contain a fluorescent reporter dye at the 5' end and a quenching dye at the 3' end. Fluorescent molecules become excited when they are irradiated and usually emit light. However, in a TaqMan probe the energy from the fluorescent dye is transferred to the quencher dye by fluorescence resonance energy transfer as long as the quencher and the dye are located in close proximity i.e. when the probe is intact. TaqMan probes are designed to anneal within a PCR product amplified by a standard PCR primer pair. If a TaqMan probe is bound to a product template, the replication of this will cause the Taq polymerase to encounter the probe. Upon doing so, the 5'exonuclease activity of the polymerase will cleave the probe. This cleavage separates the quencher and the dye, and as a result the reporter dye starts to emit fluorescence.

The TaqMan technology is used in Real-Time quantitative PCR. Since the accumulation of fluorescence mirrors the accumulation of PCR products it can can be monitored in real-time and used to quantify the amount of template initially present in the buffer.

The technology is also used to detect genetic variation such as SNP's. By designing a TaqMan probe which will specifically bind to one of two or more genetic variants it is possible to detect genetic variants by the presence or absence of fluorescence in the reaction.

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 34.3) 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 34.10) which is similar to the *Nested PCR* dialog described above (see section 34.6).

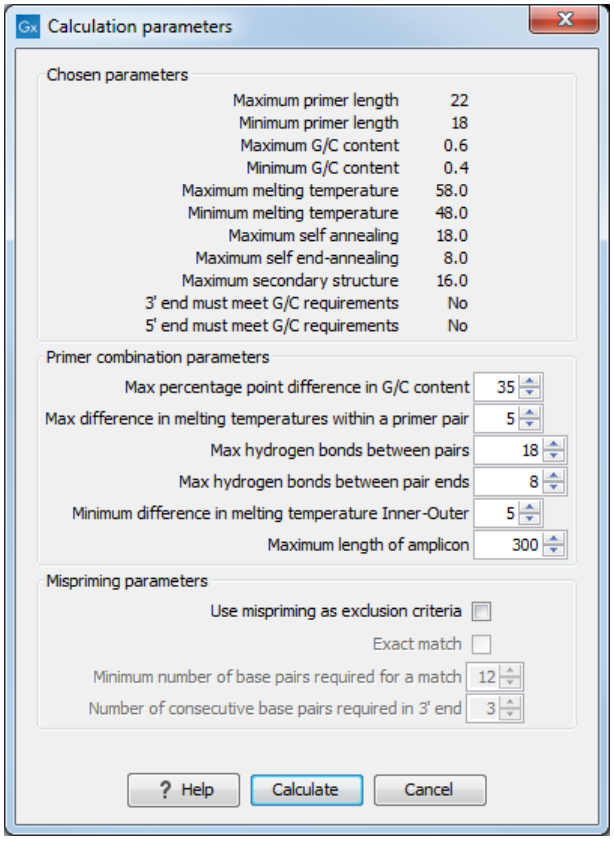


Figure 34.10: *Calculation dialog for taqman primers.*

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.

34.8 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 34.3) 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 34.11).

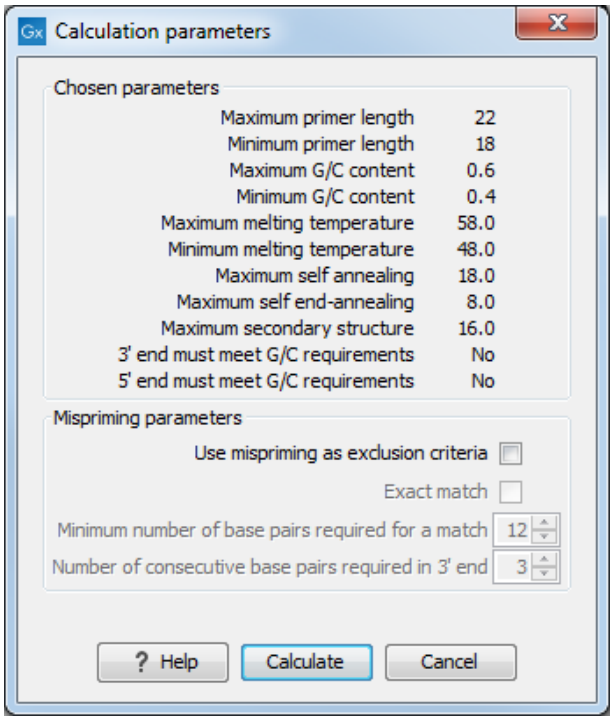


Figure 34.11: *Calculation dialog for sequencing primers.*

Since design of sequencing primers does not require the consideration of interactions between primer pairs, this dialog is identical to the dialog shown in *Standard PCR* mode when only a single primer region is chosen (see section 34.5 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.

34.9 Alignment-based primer and probe design

Biomedical Genomics Workbench allows the user to design PCR primers and TaqMan probes based on an alignment of multiple sequences.

The primer designer for alignments can be accessed with:

Toolbox | **Sanger Sequencing** ( ) | **Primers and Probes ( )**| **Design Primers ( )**

or **If the alignment is already open:** | **Click Primer Designer ( ) in the lower left part of the view**

In the alignment primer view (see figure 34.12), the basic options for viewing the template alignment are the same as for the standard view of alignments . This means that annotations such as known SNPs or exons can be displayed on the template sequence to guide the choice of primer regions.

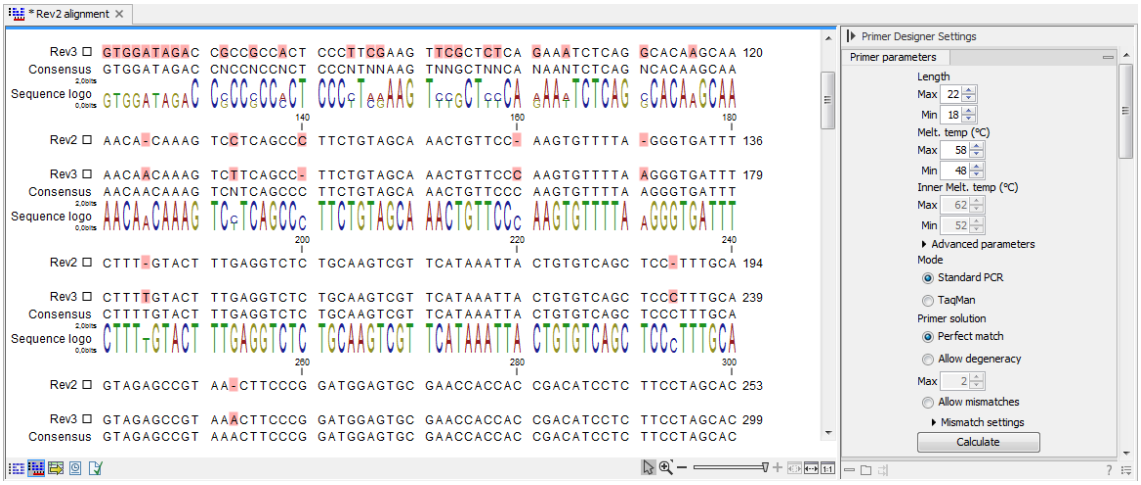


Figure 34.12: *The initial view of an alignment used for primer design.*

34.9.1 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 34.12):

- 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 34.12):

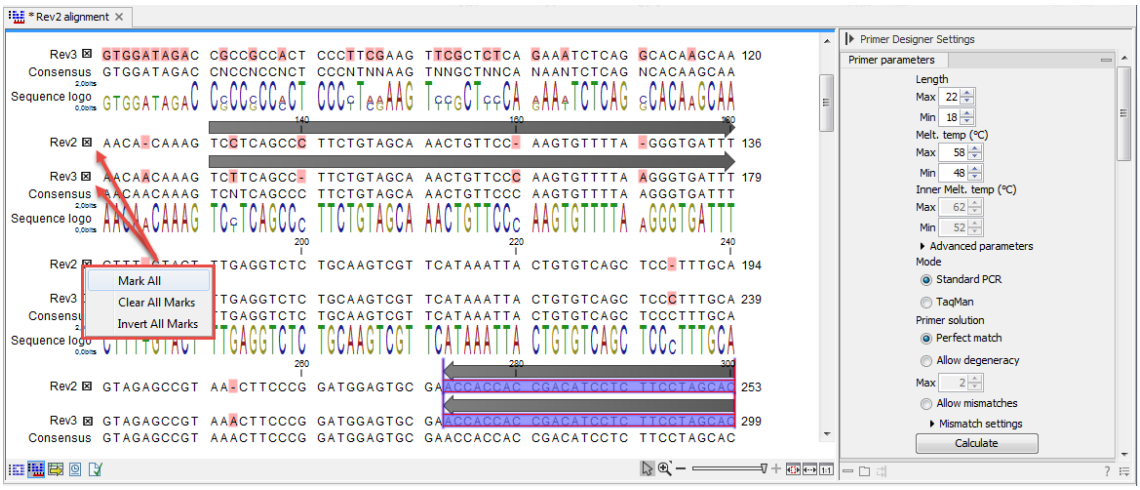


Figure 34.13: *The initial view of an alignment used for primer design.*

- 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.

34.9.2 Alignment based design of PCR primers

In this mode, a single or a pair of PCR primers are designed. *Biomedical Genomics Workbench* allows the user to design primers which will specifically amplify a group of *included* sequences but **not** amplify the remainder of the sequences, the *excluded* sequences. The selection boxes are used to indicate the status of a sequence, if the box is checked the sequence belongs to the included sequences, if not, it belongs to the excluded sequences. To design primers that are general for all primers in an alignment, simply add them all to the set of included sequences by checking all selection boxes. Specificity of priming is determined by criteria set by the user in the dialog box which is shown when the **Calculate** button is pressed (see below).

Different options can be chosen concerning the match of the primer to the template sequences in the included group:

- **Perfect match.** Specifies that the designed primers must have a perfect match to all relevant sequences in the alignment. When selected, primers will thus only be located in regions that are completely conserved within the sequences belonging to the included group.
- **Allow degeneracy.** Designs primers that may include ambiguity characters where heterogeneities occur in the included template sequences. The allowed fold of degeneracy is user defined and corresponds to the number of possible primer combinations formed by a degenerate primer. Thus, if a primer covers two 4-fold degenerate site and one 2-fold degenerate site the total fold of degeneracy is $4 * 4 * 2 = 32$ and the primer will, when supplied from the manufacturer, consist of a mixture of 32 different oligonucleotides. When scoring the available primers, degenerate primers are given a score which decreases with the fold of degeneracy.
- **Allow mismatches.** Designs primers which are allowed a specified number of mismatches to the included template sequences. The melting temperature algorithm employed includes the latest thermodynamic parameters for calculating T_m when single-base mismatches occur.

When in Standard PCR mode, clicking the **Calculate** button will prompt the dialog shown in figure 34.14.

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.

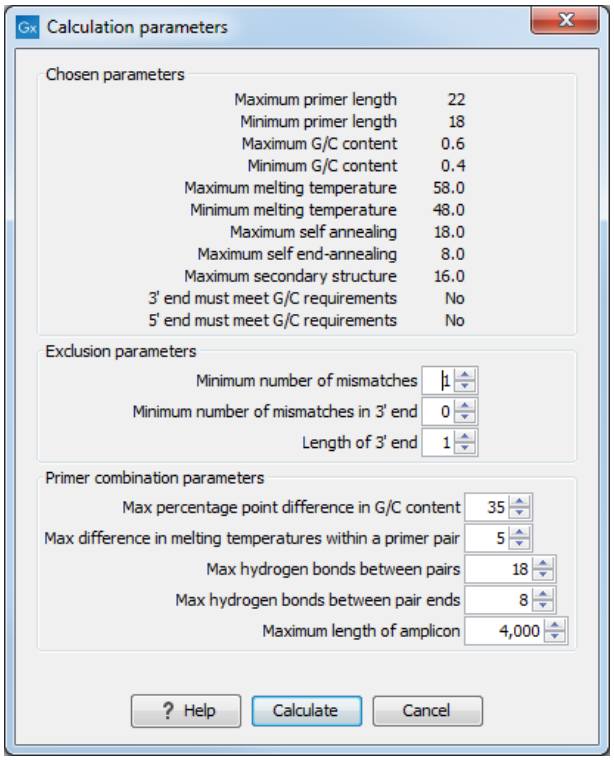


Figure 34.14: *Calculation dialog shown when designing alignment based PCR primers.*

34.9.3 Alignment-based TaqMan probe design

Biomedical Genomics Workbench allows the user to design solutions for TaqMan quantitative PCR which consist of four oligos: a general primer pair which will amplify all sequences in the alignment, a specific TaqMan probe which will match the group of *included* sequences but **not** match the *excluded* sequences and a specific TaqMan probe which will match the group of *excluded* sequences but **not** match the *included* sequences. As above, the selection boxes are used to indicate the status of a sequence, if the box is checked the sequence belongs to the included sequences, if not, it belongs to the excluded sequences. We use the terms included and excluded here to be consistent with the section above although a probe solution is presented for both groups. In TaqMan mode, primers are not allowed degeneracy or mismatches to any template sequence in the alignment, variation is only allowed/required in the TaqMan probes.

Pushing the **Calculate** button will cause the dialog shown in figure 34.15 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.

34.10 Analyze primer properties

Biomedical Genomics Workbench can calculate and display the properties of predefined primers and probes:

Toolbox | **Primers and Probes ()**| **Analyze Primer Properties ()**

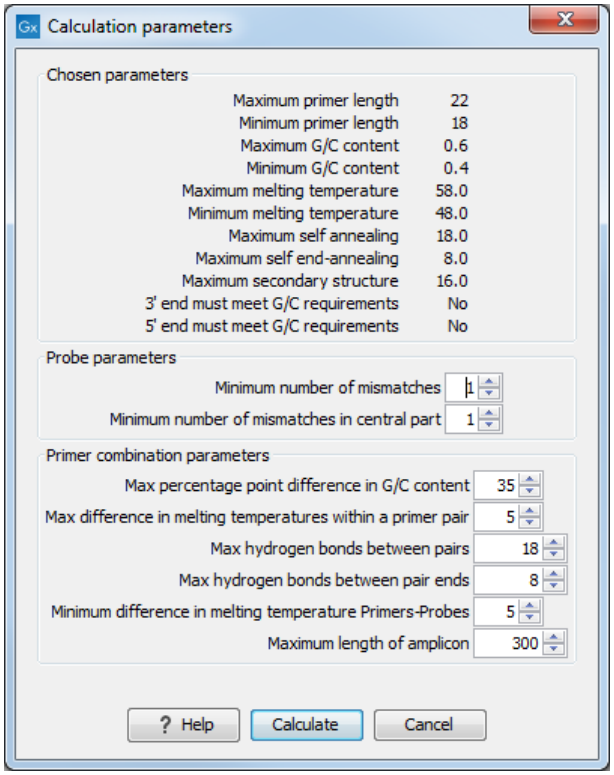


Figure 34.15: *Calculation dialog shown when designing alignment based TaqMan probes.*

If a sequence was selected before choosing the Toolbox action, this sequence is now listed in the **Selected Elements** window of the dialog. Use the arrows to add or remove a sequence from the selected elements. (Primers are represented as DNA sequences in the Navigation Area).

Clicking **Next** generates the dialog seen in figure 34.16:

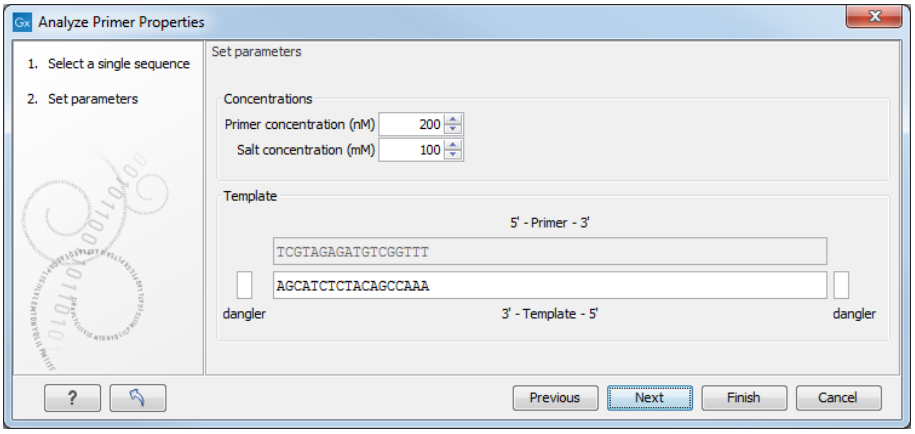


Figure 34.16: *The parameters for analyzing primer properties.*

In the *Concentrations* panel a number of parameters can be specified concerning the reaction mixture and which influence melting temperatures

- **Primer concentration.** Specifies the concentration of primers and probes in units of nanomoles (nM)
- **Salt concentration.** Specifies the concentration of monovalent cations ($[NA^+]$, $[K^+]$ and

equivalents) in units of millimoles (mM)

In the *Template panel* the sequences of the chosen primer and the template sequence are shown. The template sequence is as default set to the reverse complement of the primer sequence i.e. as perfectly base-pairing. However, it is possible to edit the template to introduce mismatches which may affect the melting temperature. At each side of the template sequence a text field is shown. Here, the dangling ends of the template sequence can be specified. These may have an important affect on the melting temperature [Bommarito et al., 2000]

Click **Finish** to start the tool. The result is shown in figure 34.17:

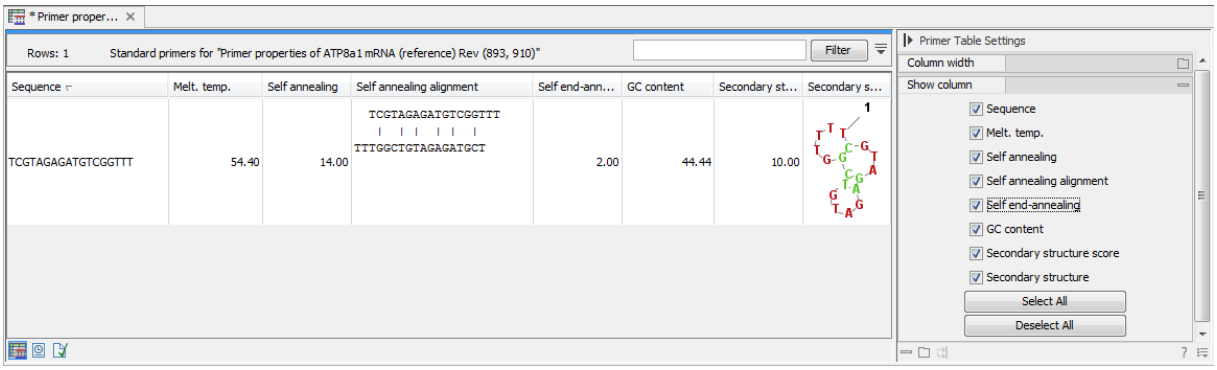


Figure 34.17: *Properties of a primer.*

In the **Side Panel** you can specify the information to display about the primer. The information parameters of the primer properties table are explained in section 34.5.3.

34.11 Find binding sites and create fragments

In *Biomedical Genomics Workbench* you have the possibility of matching known primers against one or more DNA sequences or a list of DNA sequences. This can be applied to test whether a primer used in a previous experiment is applicable to amplify 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** | **Sanger Sequencing** ( ) |**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.

34.11.1 Binding parameters

This opens the dialog displayed in figure 34.18:

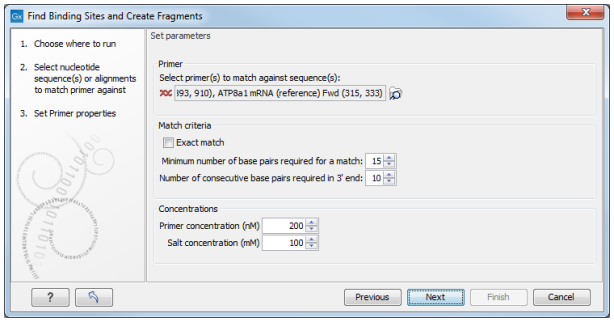


Figure 34.18: *Search parameters for finding primer binding sites.*

At the top, select one or more primers by clicking the browse ( ) button. In *Biomedical Genomics 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)

34.11.2 Results - binding sites and fragments

Specify the output options as shown in figure 34.19:

The output options are:

- **Add binding site annotations**. This will add annotations to the input sequences (see details below).

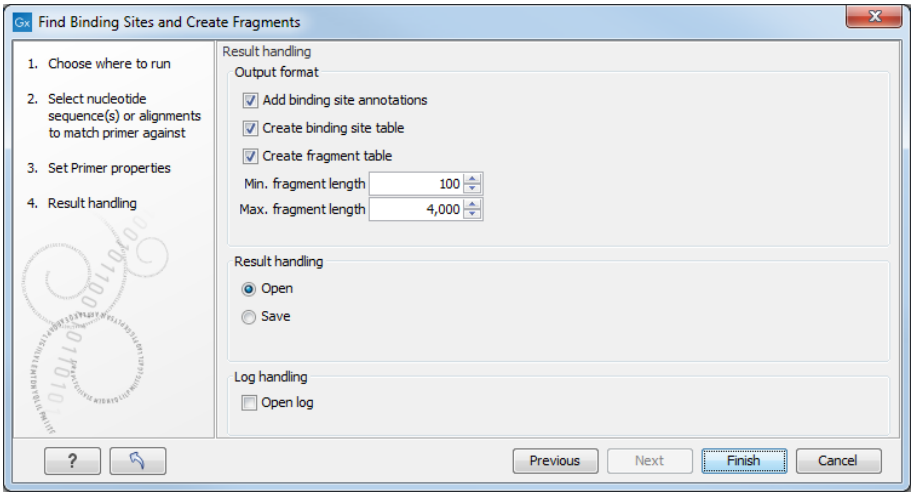


Figure 34.19: *Output options include reporting of binding sites and fragments.*

- **Create binding site table**. Creates a table of all binding sites. Described in details below.
- **Create fragment table**. Showing a table of all fragments that could result from using the primers. Note that you can set the minimum and maximum sizes of the fragments to be shown. The table is described in detail below.

Click **Finish** to start the tool.

An example of a **binding site annotation** is shown in figure 34.20.

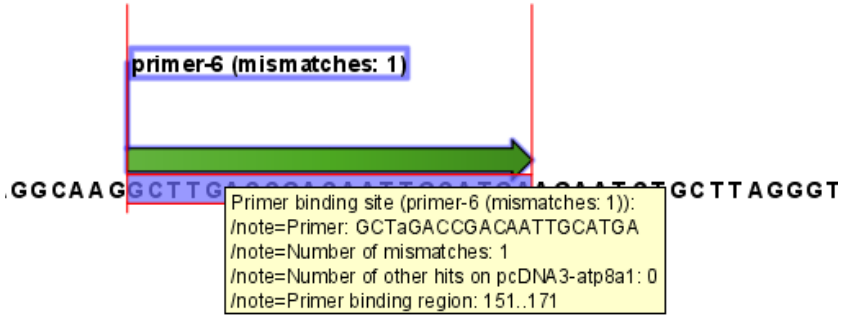


Figure 34.20: *Annotation showing a primer match.*

The annotation has the following information:

- **Sequence of the primer**. Positions with mismatches will be in lower-case (see the fourth position in figure 34.20 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.

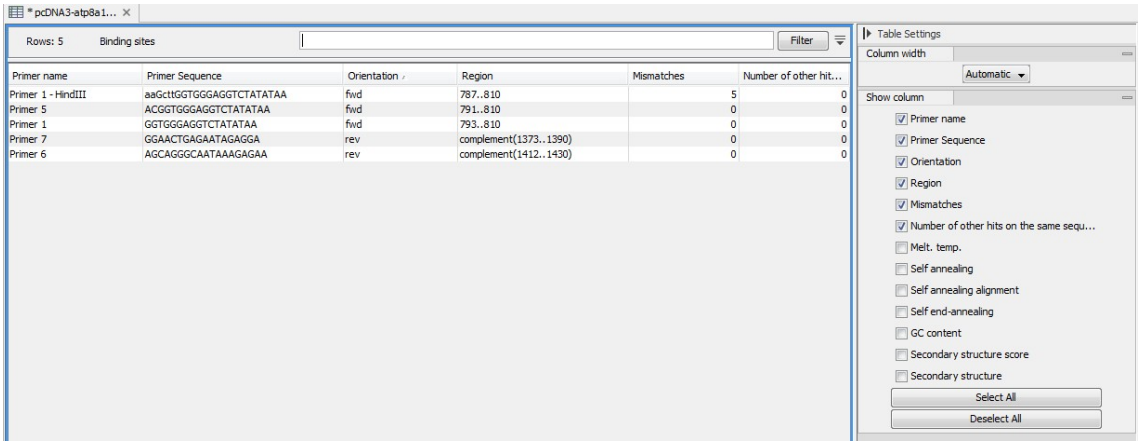


Figure 34.21: *A table showing all binding sites.*

An example of the **primer binding site table** is shown in figure 34.21.

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 34.5.3. You can use this table to browse the binding sites. If you make a split view of the table and the sequence (see section 2.1.6), you can browse through the binding positions by clicking in the table. This will cause the sequence view to jump to the position of the binding site.

An example of a **fragment table** is shown in figure 34.22.

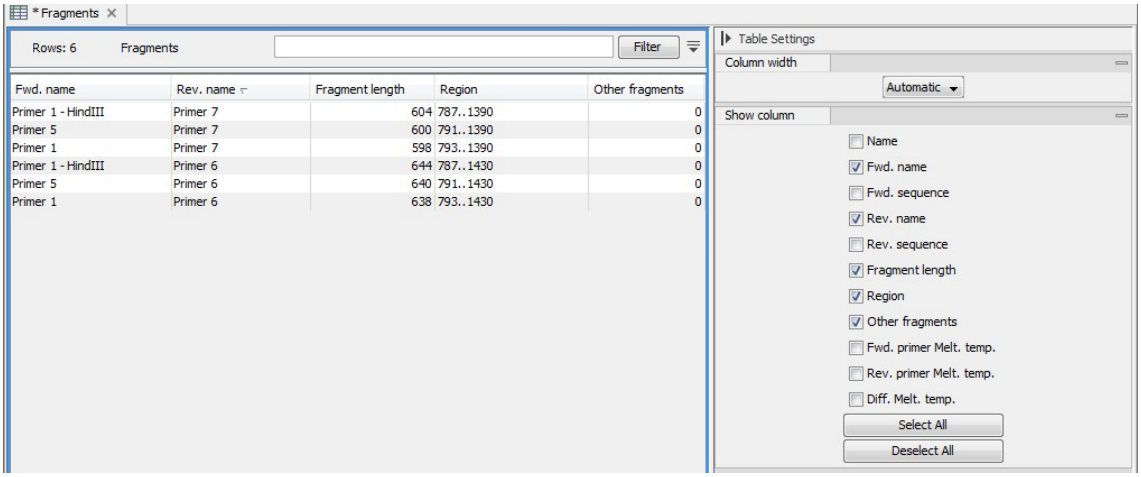


Figure 34.22: *A table showing all possible fragments of the specified size.*

The table first lists the names of the forward and reverse primers, then the length of the fragment and the region. The last column tells if there are other possible fragments fulfilling the length criteria on this sequence. This information can be used to check for competing products in the PCR. In the **Side Panel** you can show information about melting temperature for the primers as well as the difference between melting temperatures.

You can use this table to browse the fragment regions. If you make a split view of the table and the sequence (see section 2.1.6), you can browse through the fragment regions by clicking in the table. This will cause the sequence view to jump to the start position of the fragment.

There are some additional options in the fragment table. First, you can annotate the fragment on

the original sequence. This is done by right-clicking (Ctrl-click on Mac) the fragment and choose **Annotate Fragment** as shown in figure 34.23.

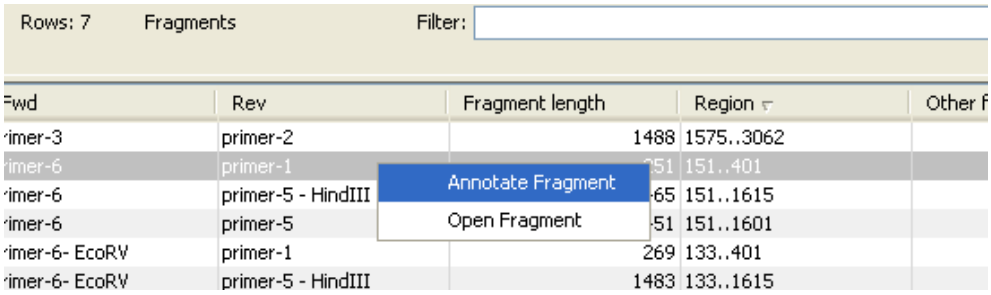


Figure 34.23: *Right-clicking a fragment allows you to annotate the region on the input sequence or open the fragment as a new sequence.*

This will put a *PCR fragment* annotations on the input sequence covering the region specified in the table. As you can see from figure 34.23, 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.

34.12 Order primers

To facilitate the ordering of primers and probes, *Biomedical Genomics Workbench* offers an easy way of displaying and saving a textual representation of one or more primers:

Toolbox | **Sanger Sequencing** ( ) | **Primers and Probes ( )**| **Order Primers ( )**

This opens a dialog where you can choose additional primers. Clicking **OK** opens a textual representation of the primers (see figure 34.24). 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. The created object can also be saved and exported as a text file.

See figure 34.24

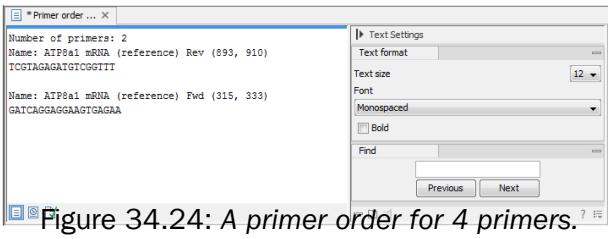


Figure 34.24: *A primer order for 4 primers.*

Part XI

Epigenomics Analysis

Chapter 35

Epigenomics

Contents

35.1 ChIP-Seq Analysis

Chromatin immunoprecipitation followed by high-throughput sequencing (ChIP-Seq) is a method for analyzing DNA-protein interactions. Peak finding is an essential post-processing step in the analysis and interpretation of ChIP-Seq data and many tools that have been developed specifically for this purpose.

Most peak callers are difficult to parametrize such that they accurately discriminate active and inactive promoter regions [Rye et al., 2011, Heydarian et al., 2014]. Nevertheless, these peaks are clearly visible to the human eye since they show a distinct shape. The Transcription Factor ChIP-Seq tool takes a different approach, which is conceptually intuitive and modular by learning the shape of the signal from the data. The parametrization is done by giving positive and negative examples of peak shapes, making the parametrization process explicit and easily understandable.

The Transcription Factor ChIP-Seq tool uses this approach to identify genomic regions with significantly enriched read coverage and a read distribution with a characteristic shape.

ChIP-Seq data analysis is typically based on identification of genomic regions where the signal (i.e. number of mapped reads) is significantly enriched. The detection of the enrichment is based on a background model or the comparison with a ChIP-Seq sample where the immunoprecipitation step is omitted. The shape of the signal from ChIP-Seq data depends on which protein was targeted in the immunopreciptation reaction [Stanton et al., 2013, Kumar et al., 2013]. For example, the typical signal shape of a transcription factor binding site like NRSF shows a high concentration of forward reads followed by a high concentration of reverse reads (figure 35.1).

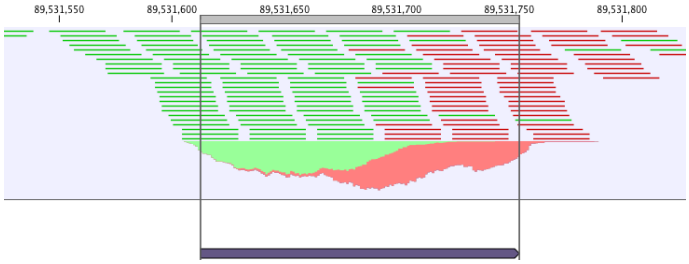


Figure 35.1: *Distribution of forward (green) and reverse (red) reads around a binding site of the transcription factor NRSF.*

The tool makes use of this characteristic shape to identify enriched regions (peaks) in ChIP-Seq data.

35.1.1 Quality Control of ChIP-Seq data

During the first step of the analysis, the Transcription Factor ChIP-Seq tool analyzes the input to check if the input data satisfy the assumptions made by the algorithm and to compute several quality measures. The cross-correlation between reads mapping in the forward and in the reverse strand is often used to investigate the quality of ChIP-Seq experiments [Landt et al., 2012, Marinov et al., 2014]. The quality is determined with respect to the two main peaks of the cross-correlation plot (figure 35.2), the peak at the read length (often called a phantom peak [Landt et al., 2012]) and the one at the fragment length. The peak at the fragment length is typically higher than the peak at the read length and the background (figure 35.2) for successful ChIP-Seq experiments.

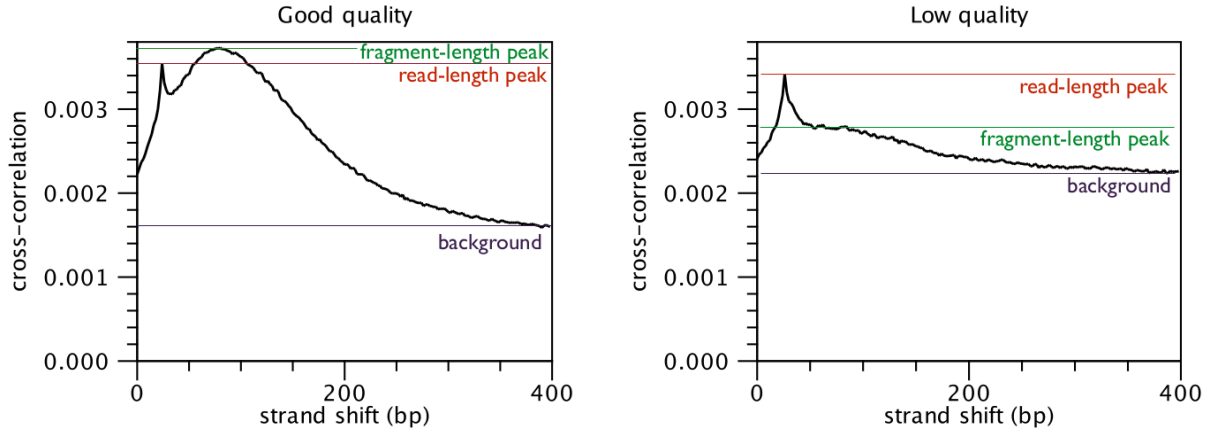


Figure 35.2: *Difference in cross-correlation profiles in ChIP experiments of good and low quality.*

For each input file of the analysis, the Transcription Factor ChIP-Seq tool calculates and reports several quality measures. Those quality measures have been investigated by the modENCODE consortium and are described in more detail in [Landt et al., 2012]. The quality measures are:

Number of mapped reads For mammalian cells (e.g. human and mouse), this value should be at least 10 million reads. For smaller organisms such as worm and fly, this value should be at least 2 million reads.

Normalized strand coefficient The normalized strand coefficient describes the ratio between the fragment-length peak and the background cross-correlation values. This value should be greater than 1.05 for ChIP-Seq experiments.

Relative strand correlation The relative strand correlation describes the ratio between the fragment-length peak and the read-length peak in the cross-correlation plot. This value should be high (at least 0.8) for transcription factor binding sites, which have a concentrated signal. However, this value can be low even for successful ChIP-Seq experiments on histone modifications [Landt et al., 2012].

35.1.2 Learning peak shapes

In order to learn a characteristic shape from ChIP-Seq data, the Transcription Factor ChIP-Seq tool analyzes the genomic coverage of the reads. For each read mapping, the 5' position of the reads mapping in the forward strand and the 3' position of the reads mapping in the reverse strand are extracted. Those values are then normalized to create a coverage value for the forward and the reverse strand. In order to learn the characteristic peak shape of ChIP-Seq data, the Transcription Factor ChIP-Seq tool identifies a set of positive regions, i.e. regions with very apparent peaks. Those regions are easy to find and are typically found by every peak-caller. The Transcription Factor ChIP-Seq tool identifies these regions by finding areas with very high coverage in the ChIP-Seq data. The average shape of the positive regions of the NRSF transcription factor is shown in for the forward and reverse strand in figure 35.3.

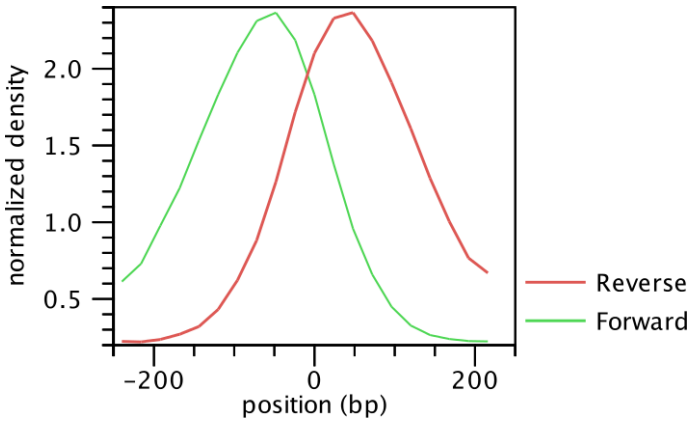


Figure 35.3: *Average peak shape of the transcription factor NRSF.*

Next, the Transcription Factor ChIP-Seq tool builds a filter, which can be used to identify genomic regions whose read coverage profile matches the characteristic peak shape and to determine the statistical significance of this match. In order to build such a filter, examples of positive (e.g. ChIP-Seq peaks) and negative (e.g. background noise, PCR artifacts) profiles are needed as input. The Transcription Factor ChIP-Seq tool uses regions with very high coverage in the experiment ChIP-Seq as positive examples. If control ChIP-Seq experiments are given, regions with high coverage in the control and low in the experimental ChIP-Seq data are used as negative examples, as they are probably originated from PCR artifacts. If there is no information to build a negative profile from, the profile is estimated from the sequencing noise.

Once the positive and negative regions have been identified, the Transcription Factor ChIP-Seq tool learns a filter that matches the average peak shape, which we term peak shape filter. The filter implemented is called Hotelling Observer and was chosen because it is the matched filter that maximizes the AUCROC (Area Under the Curve of the Receiver Operator Characteristic), one of the most widely used measures for algorithmic performance.

The Hotelling observer h is defined as:

$$h = \left(\frac{R_p + R_n}{2}\right)^{-1} (\mathbb{E}[X_p] - \mathbb{E}[X_n]), \tag{35.1}$$

where $\mathbb{E}[X_p]$ is the average profile of the *positive* regions, $\mathbb{E}[X_n]$ is the average profile of the *negative* regions, while R_p and R_n denote the covariance matrices between the *positive* and *negative* profiles, respectively. The Hotelling Observer has already previously been successfully used for calling ChIP-Seq peaks [Kumar et al., 2013]. An example of Hotelling observer is shown in figure 35.4.

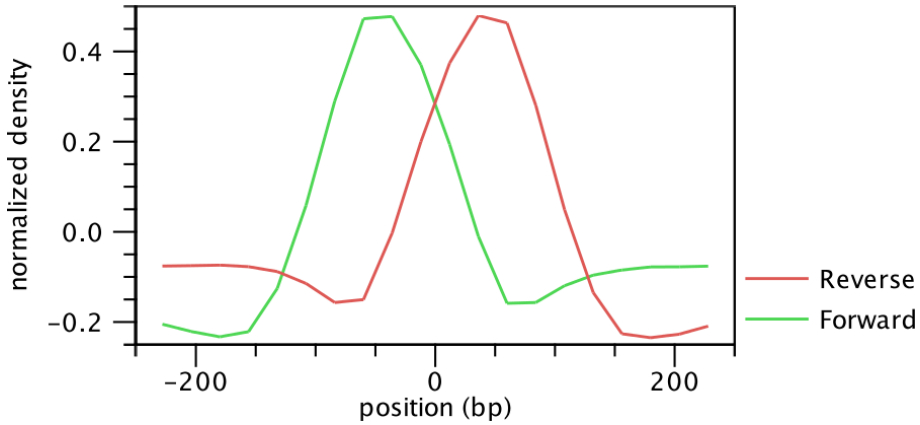


Figure 35.4: *Peak shape filter for the transcription factor NRSF.*

35.1.3 Applying peak shape filters to call peaks

The peak shape filter is then applied to the experimental data and a score is calculated at each genomic position. The score is obtained by extracting the genomic coverage profile of a window centered at the genomic position and then comparing this profile to the peak shape filter. The result of this comparison is defined as peak shape score. The peak shape score is standardized and follows a standard normal distribution, so a p-value for each genomic position can be calculated as p-value $= \Phi(-\text{Peak shape score of the peak center})$, where Φ is the standard normal cumulative distribution function.

Once the peak shape score for the complete genome is calculated, peaks are identified as genomic regions where the maximum peak shape score is greater than a given threshold. The center of the peak is then identified as the genomic region with the highest peak shape score and the boundaries are determined by the genomic positions where the peak shape score becomes negative.

35.1.4 Running the Transcription Factor ChIP-Seq tool

To run the Transcription Factor ChIP-Seq tool:

Toolbox | **Epigenomics Analysis** ( ) | **Transcription Factor ChIP-Seq** ( )

This will open up the wizard shown in figure 35.5 where you can select the input data (the mapped ChIP-Seq reads). Multiple inputs (e.g. replicate experiments) are accepted, provided that they refer to the same genome. Track based read mappings ( ) and stand-alone read mappings ( ) / ( ) are both accepted.

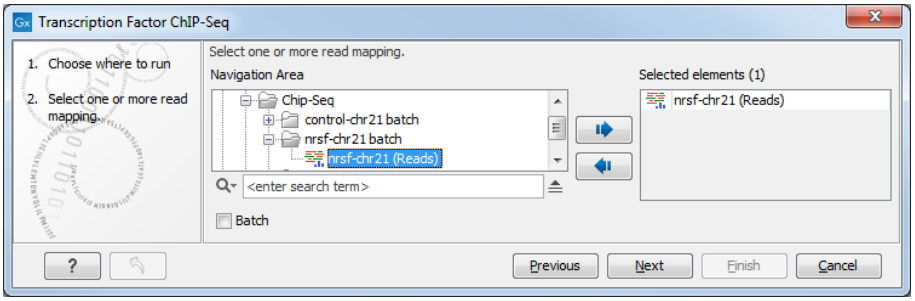


Figure 35.5: *Select the input data for the Transcription Factor ChIP-Seq tool.*

Click on the button labeled **Next** to go to the next wizard step (shown in figure 35.6).

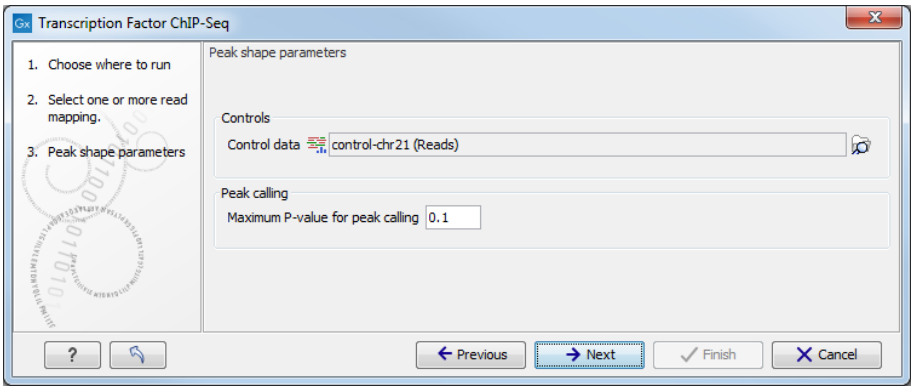


Figure 35.6: *Options for the Transcription Factor ChIP-Seq tool.*

In this wizard step you have the following options:

- **Control data** The control data, typically a ChIP-Seq sample where the immunoprecipitation step is omitted, can be specified in this option.
- **Maximum P-value for peak calling** The threshold for reporting peaks can be specified by this option.

Click on the button labeled **Next** to go to the wizard step shown in figure 35.7.

In addition to the annotation track with **Peak annotations** ( ) that will always be generated by the algorithm, you can choose to select additional output types.

The options are:

- **QC report** ( ) Generates a quality control report that allows you to check the quality of the reads. The QC report contains metrics about the quality of the ChIP-Seq experiment. It lists the number of mapped reads, the normalized strand coefficient, and the relative strand correlation for each mapping. For each metric, the **Status** column will be **OK** if the experiment has good quality or **Low** if the metric is not as high as expected. Furthermore, the QC report will show the mean read length, the inferred fragment length, and the window size that the algorithm would need to be able to model the signal shape. In case the input contains paired-end reads, the report will also contain the empirical fragment length distribution. The metrics and their definitions are described in more detail in section 35.1.1.

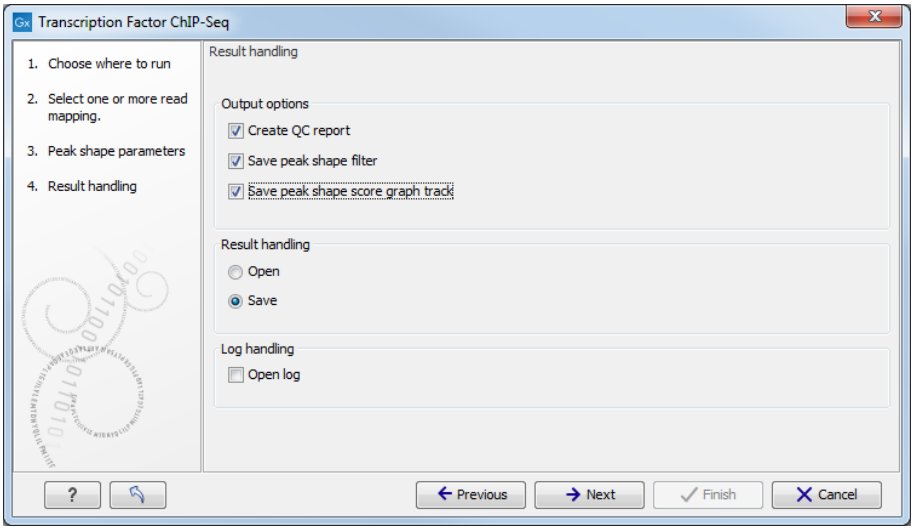


Figure 35.7: *Output options for the Transcription Factor ChIP-Seq tool.*

- **Peak shape filter** ( ) The peak shape filter contains the Hotelling Observer filter that was learned by the Transcription Factor ChIP-Seq algorithm. For the definition of Peak shape, see section 35.1.3.
- **Peak shape score ( )** A graph track containing the peak shape score. The track shows the peak shape score for each genomic position. To save disk space, only peak shape scores greater than zero are reported. For the definition of peak shape score, see section 35.1.3.

Choose whether you want to open the results directly, or save the results in the **Navigation Area**. If you choose to save the results, you will be asked to specify where you would like to save them.

Peak track

The main result of the Transcription Factor ChIP-Seq algorithm is an annotation track containing the peaks. For each peak the following quantities are reported in the table, which can be opened by clicking on the table icon ( ) in the lower left corner of the peak annotation track. For more details on some of the values included in the table, see section 35.1.3.

- **Chromosome** The chromosome where the peak is located.
- **Region** The position of the peak.
- **Center of peak** The center position of the peak. This is determined as the genomic position that matches the peak shape filter best.
- **Length** The length of the peak.
- **Peak shape score** The peak shape score of the peak.
- **P-value** The p-value of the peak.

The peak annotation track is most informative when combined with the read mapping in a Track List (you can see how to create a track list here: `http://resources.qiagenbioinformatics.`

com/manuals/clcgenomicsworkbench/current/index.php?manual=Track_lists.html). A Track List containing the mapped reads, the Peak track, and the Peak shape score track is shown in figure 35.8.

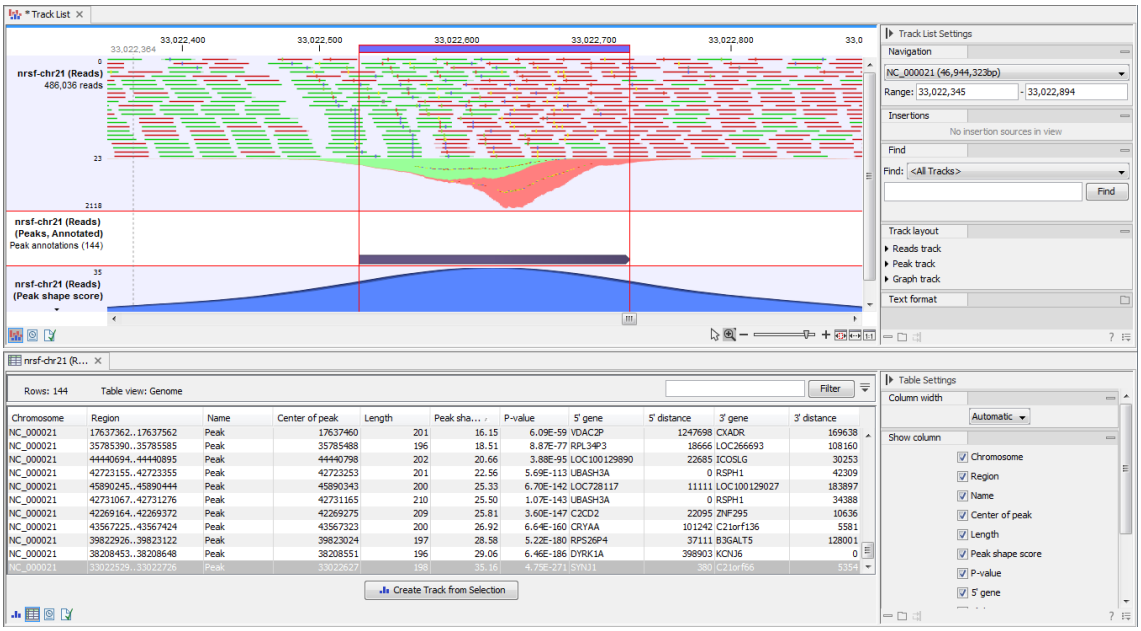


Figure 35.8: *Inspection of the result of the Transcription Factor ChIP-Seq tool.*

Note that if you make a split view of the table and the peak annotation track (by holding down the Ctrl key (Cmd on Mac) while clicking on the table icon ( ) in the lower left corner of the peak annotation track), you will be able to browse through the peaks by clicking in the table, as the peak annotation track and the table are connected. As a result the view will jump to the position of the peak selected in the table.

35.2 Annotate with nearby gene information

This tool will create a copy of the annotation track ( ) used as input and add information about nearby genes.

Toolbox | **Epigenomics Analysis ( )** | **Annotate with Nearby Gene Information ( )**

First, select the track you wish to annotate and click **Next**. The tool was designed for ChIP-Seq analysis, but you can choose any kind of annotation track as input. Next, select a gene track with a compatible genome (figure 35.9).

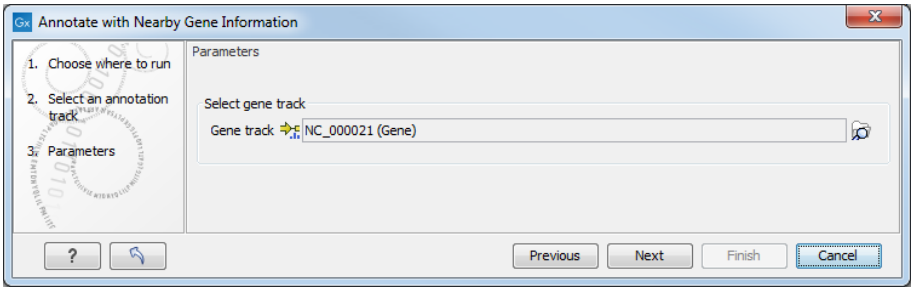


Figure 35.9: *Select gene track.*

The result of this tool is a new annotation track with all the annotations from the input track and

with additional information about nearby genes and four columns will be added to the table view:

- **5' gene** The name of the nearest upstream gene.
- **5' distance** The distance from the nearest upstream gene or 0 if the feature overlaps the nearest gene. The distance value is determined by the shortest distance between an end of the gene annotation and an end of the peak annotation, regardless of the annotation orientation (see figure 35.10).
- **3' gene** The name of the nearest downstream gene.
- **3' distance** The distance from the nearest downstream gene or 0 if the feature overlaps the nearest downstream gene. The distance value is determined by the shortest distance between an end of the gene annotation and an end of the peak annotation, regardless of the annotation orientation.

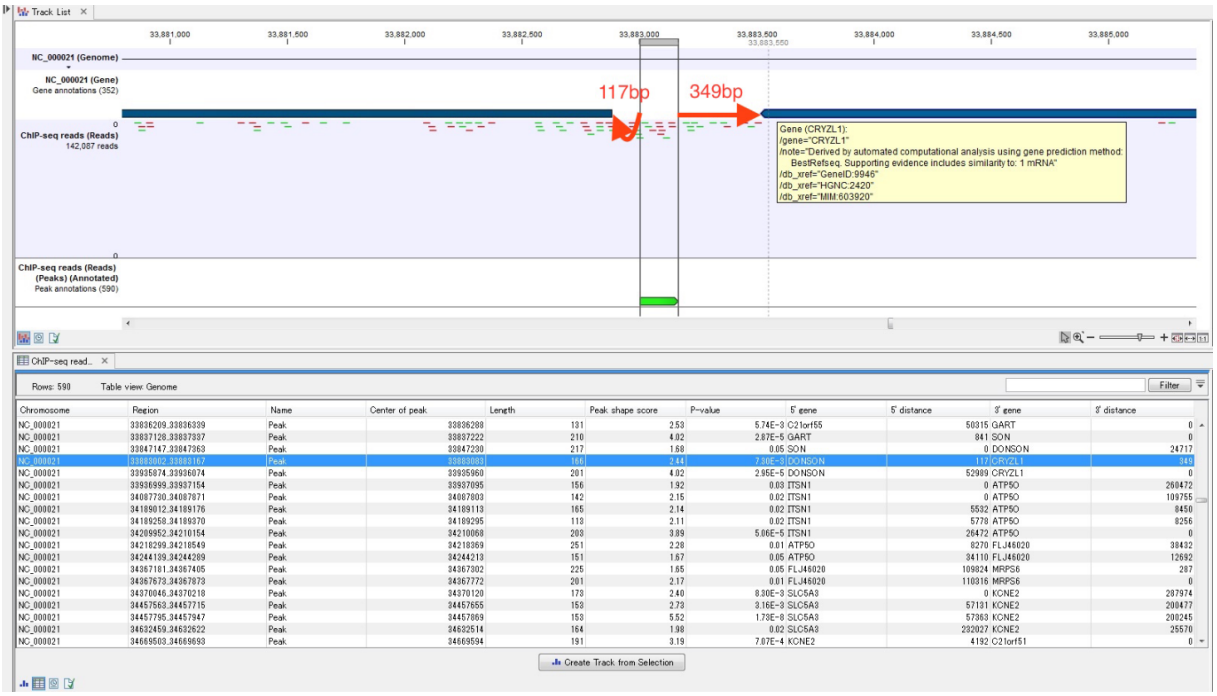


Figure 35.10: *The figure is a split view between a table view and a track list showing the gene annotation track, the ChIP-Seq reads, and the annotated ChIP peaks.The red arrows and numbers illustrate the 5' distance and 3' distance.*

Chapter 36

Legacy tools

Contents

A folder with the name **Legacy Tools** () is found at the bottom of the toolbox. The tools found in this folder are tools that have been used in older versions of the workbench but are now deprecated. These tools will be retired in a future version of the workbench.

If you are using tools in the Legacy section of the Toolbox in workflows, then we recommend redesigning those workflows to exclude these tools. In the cases where a tool has been deprecated and a new tool introduced to take its place, please try updating the workflows to include the new tool.

If you have concerns about the retirement of particular tools currently appearing in the Legacy section of the Toolbox, please contact QIAGEN Bioinformatics Technical Service team at ts-bioinformatics@qiagen.com.

36.1 Import Roche 454

Choosing the Roche 454 import will open the dialog shown in figure 36.1.

We support import of two kinds of data from 454 GS FLX systems:

- Flowgram files (`.sff`) which contain both sequence data and quality scores amongst others. However, the flowgram information is currently not used by *Biomedical Genomics Workbench*. There is an extra option to make use of clipping information (this will remove parts of the sequence as specified in the .sff file).
- Fasta/qual files:
  - 454 FASTA files (`.fna`) which contain the sequence data.
  - Quality files (`.qual`) which contain the quality scores.
  - Compressed data in gzip format is also supported (.gz).

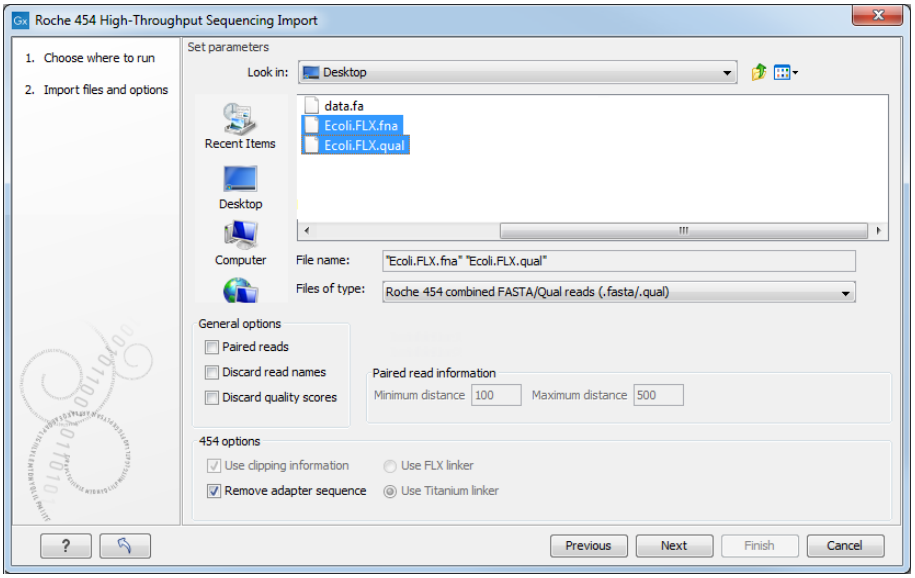


Figure 36.1: *Importing data from Roche 454.*

The **General options** to the left are:

- **Paired reads**. The paired protocol for 454 entails that the forward and reverse reads are separated by a linker sequence. During import of paired data, the linker sequence is removed and the forward and reverse reads are separated and put into the same sequence list (their status as forward and reverse reads is preserved). You can change the linker sequence in the **Preferences** (in the **Edit** menu) under **Data**. Since the linker for the FLX and Titanium versions are different, you can choose the appropriate protocol during import, and in the preferences you can supply a linker for both platforms (see figure 36.2. Note that since the FLX linker is palindromic, it will only be searched on the plus strand, whereas the Titanium linker will be found on both strands. Some of the sequences may not have the linker in the middle of the sequence, and in that case the partial linker sequence is still removed, and the single read is put into a separate sequence list. Thus when you import 454 paired data, you may end up with two sequence lists: one for paired reads and one for single reads. Note that for de novo assembly projects, only the paired list should be used since the single reads list may contain reads where there is still a linker sequence present but only partially due to sequencing errors. Read more about handling paired data in section 6.3.7.

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.

- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. If you have selected the fna/qual option and choose to discard quality scores, you do not need to select a .qual file.

Note! During import, partial adapter sequences are removed (`TCAG` and `ATGC`), and if the full sequencing adapters `GCCTTGCCAGCCCGCTCAG`, `GCCTCCCTCGCGCCATCAG` or their reverse

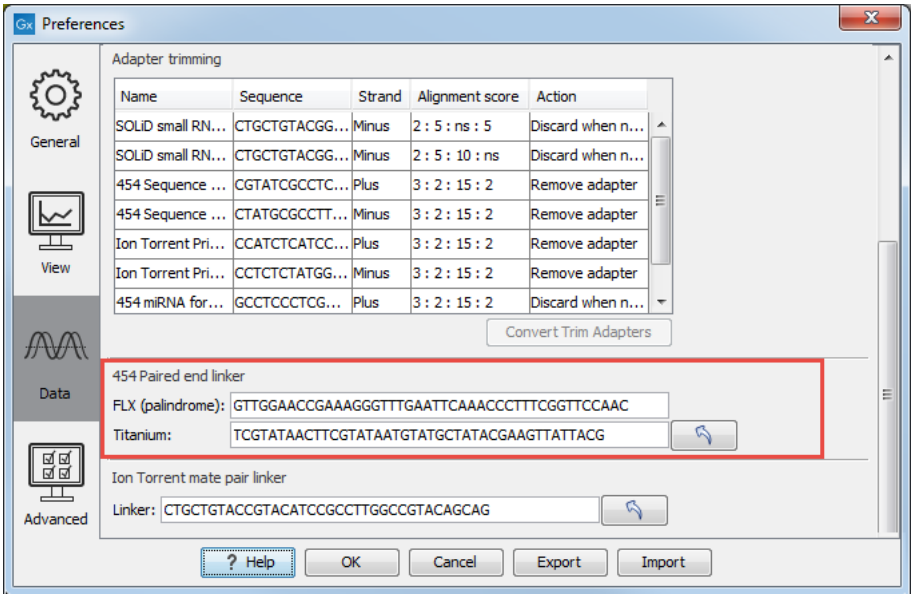


Figure 36.2: *Specifying linkers for 454 import.*

complements are found, they are also removed (including tailing Ns). If you do not wish to remove the adapter sequences (e.g. if they have already been removed by other software), please uncheck the **Remove adapter sequence** option.

Click **Next** to adjust how to handle the results (see section 8.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 8.3).

36.2 Import SOLiD

Choosing the SOLiD import will open the dialog shown in figure 36.3.

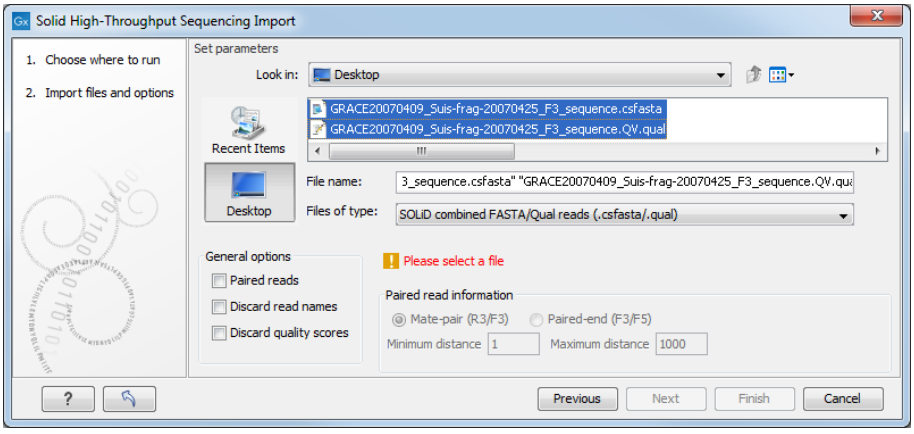


Figure 36.3: *Importing data from SOLiD.*

There are two formats accepted: the XSQ format which is the native format of newer SOLiD systems, and the csfasta format which is the color space version of fasta format.

The XSQ format

An XSQ file can contain results from multiple libraries produced from the same sequencing run. These are identified by a barcode on each read, and when the XSQ file is produced, each read is placed into its appropriate library based on its barcode. The XSQ importer creates separate sequence lists for each library.

Sometimes when an XSQ file is produced, a barcode cannot be identified accurately enough to place the read into a specific library, or the read is for some other reason not assigned to a library. In this case, the read is placed into an "Unclassified" or "Unassigned" library.

In the case of paired reads, it sometimes happens that one read of a pair could not be read. When the XSQ file is imported in the *Biomedical Genomics Workbench*, the other read of such a pair is placed into a sequence list with " (single)" appended to the name, whereas all intact pairs are placed (alternating) into a sequence list with " (paired)" appended to the name. Thus, two sequence lists are produced for the library.

Hence, when importing data in XSQ format the number of imported files can vary. In the example shown here, where the XSQ file contain a library with the name "Main" (containing paired reads) and an "Unclassified" library (containing reads where e.g. the barcode could not be read), the imported data are segregated into the following sequence lists:

1. Main (single)
2. Main (paired)
3. Unclassified

An XSQ file sometimes contains reads in both base space and color space, and when that is the case, each read library in the XSQ file that contains reads in both formats will result in two files, with " (base space)" and " (color space)" appended to their names, respectively.

The csfasta format

If you want to import quality scores with csfasta files, qual files should also be provided. The reads in a csfasta file look like this:

```
>2_14_26_F3
T0112131222002211230321112210212101313322221O1
>2_14_192_F3
T1100212211003100301200220322211132102211222З
>2_14_233_F3
T0110013323111212123120223102033122011321112З3
>2_14_294_F3
T213012132300000021323212232.03300033102330332
```

All reads start with a T which specifies the right phasing of the color sequence.

If a reads has a `.` as you can see in the last read in the example above, it means that the color calling was ambiguous (this would have been an `N` if we were in base space). In this case, the Workbench simply cuts off the rest of the read, since there is no way to know the right phase of the rest of the colors in the read. If the read starts with a dot, it is not imported. If all reads start with a dot, a warning dialog will be displayed. The handling of dots is identical for XSQ and csfasta files.

In the quality file, the equivalent value is `-1`, and this will also cause the read to be clipped.

When the example above is imported into the Workbench, it looks as shown in figure 36.4.

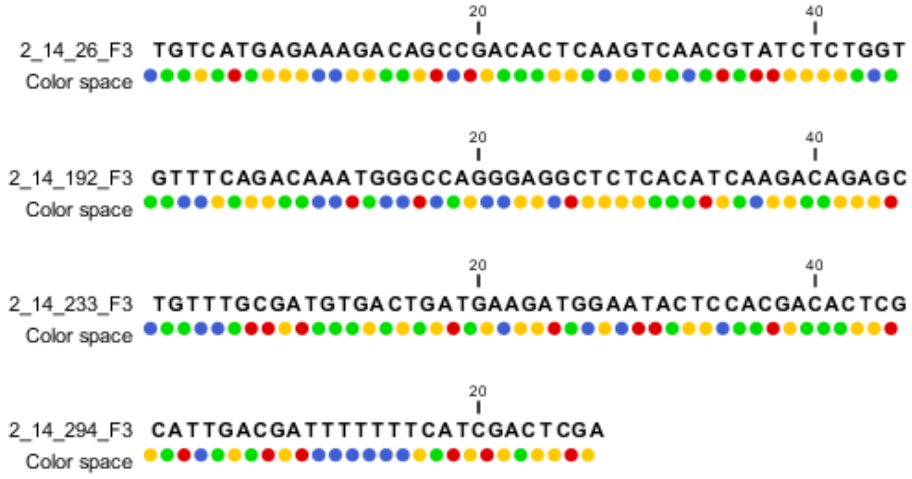


Figure 36.4: *Importing data from SOLiD. Note that the fourth read is cut off so that the color following the dot are not included*

For more information about color space, please see section 22.4.

In addition to the csfasta and XSQ formats used by SOLiD, you can also input data in fastq format. This is particularly useful for data downloaded from the Sequence Read Archive at NCBI (`http://www.ncbi.nlm.nih.gov/Traces/sra/`). An example of a SOLiD fastq file is shown here with both quality scores and the color space encoding:

```
@SRR016056.1.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_130.1 length=50
T3100031312131021102231222331121211302212120133221 3
+SRR016056.1.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_130.1 length=50
!*%;2'%%050%'0'3%%5*.%%%),%%%%&%%%%%%'%%%%%'%%3+%%%
@SRR016056.2.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_223.1 length=50
T2000220112002121101201033221112213321233122130222 2
+SRR016056.2.1 AMELIA_20071210_2_YorubanCGB_Frag_16bit_2_51_223.1 length=50
!%%)%'))'&'%(((&%/&)%+(%%%&%%%%%%%%%%%%%%%%+%%%%%%+'
```

Compressed data in gzip format is also supported (.gz).

The **General options** to the left are:

- **Paired reads**. When you import paired data, the Workbench expects the first reads of the pairs to be in one file and the second reads of the pairs to be in another. When importing one pair of files, the first file in a pair will is assumed to contain the first reads of the pair, and the second file is assumed to contain the second read in a pair. Two different protocols are supported:
  - **Mate-pair**. For mate-pair data, the reads should be in two files with `_F3` and `_R3` in front of the file extension. The orientation of the reads is expected to be forward-forward.

  - **Paired-end**. For paired-end data, the reads should be in two files with `_F3` and `_F5-P2` or `_F5-BC`. The orientation is expected to be forward-reverse.

  Read more about handling paired data in section 6.3.7. Please note that for XSQ files, the pairing protocol is defined in the file itself, which means that the choices of protocol will be ignored.

  An example of a complete list of the four files needed for a SOLiD mate-paired data set including quality scores:

```
dataset_F3.csfasta   dataset_F3.qual
dataset_R3.csfasta   dataset_R3.qual
```

  or

```
dataset_F3.csfasta   dataset_F3_.QV.qual
dataset_R3.csfasta   dataset_R3_.QV.qual
```

- **Discard read names**. For high-throughput sequencing data, the naming of the individual reads is often irrelevant given the huge amount of reads. This option allows you to discard this option to save disk space.
- **Discard quality scores**. Quality scores are visualized in the mapping view and they are used for SNP detection. If this is not relevant for your work, you can choose to **Discard quality scores**. One of the benefits from discarding quality scores is that you will gain a lot in terms of reduced disk space usage and memory consumption. If you choose to discard quality scores, you do not need to select a .qual file when importing csfasta files.

Click **Next** to adjust how to handle the results (see section 8.2). We recommend choosing **Save** in order to save the results directly to a folder, since you probably want to save anyway before proceeding with your analysis. There is an option to put the import data into a separate folder. This can be handy for better organizing subsequent analysis results and for batching (see section 8.3).

Part XII

Appendix

Appendix A

Use of multi-core computers

Many tools in CLC Workbenches and Servers can make use of multi-core CPUs. This does not necessarily mean that all available CPU cores are used throughout the analysis. It means that these tools benefit from running on computers with multiple CPU cores.

Tools available differ between CLC Workbenches. In the table, the availability of these tools in different CLC Workbench Toolbox menus is indicated with an X.

Use of multi-core computers	Genomics	Biomedical Genomics
Add Conservation Scores		X
Add Information about Amino Acid Changes		X
Add Information from Variants		X
Add Exon Number		X
Add Flanking Sequence		X
QC for Read Mapping		X
Amino Acid Changes	X	
Annotate and Merge Counts	X	X
Annotate from Known Variants	X	
Annotate with Conservation Score	X	
Annotate with Exon Numbers	X	
Annotate with Flanking Sequences	X	
Basic Variant Detection	X	X
BLAST (will not scale well on many cores)	X	
Compare Sample Variant Tracks	X	
Copy Number Variant Detection		X
Create Alignment	X	X
Create Detailed Mapping Report	X	
Create Sequencing QC Report	X	
Create Statistics for Target Regions	X	
De Novo Assembly	X	
Dock Ligands		
Download Reference Genome Data	X	
Extract and Count	X	X
Filter against Control Reads	X	
Filter against Known Variants	X	
Filter Marginal Variant Calls	X	
Filter Reference Variants	X	
Fisher Exact Test	X	
Fixed Ploidy Variant Detection	X	X
GO Enrichment Analysis	X	
Identify Enriched Variants in Case vs Control Group		X
Identify Highly Mutated Gene Groups and Pathways		X
Identify Variants with Effect on Splicing		X
Import Molecules from SMILES or 2D		
InDels and Structural Variants	X	X
K-mer Based Tree Construction	X	
Link Variants to 3D Protein Structure	X	X
Local Realignment	X	X
Low Frequency Variant Detection	X	X
Map Reads to Contigs	X	
Map Reads to Reference	X	X
Maximum Likelihood Phylogeny	X	
Merge Annotation Tracks	X	
Model Testing	X	
Predict Splice Site Effect	X	

Use of multi-core computers	Genomics	Biomedical Genomics
Probabilistic Variant Detection (legacy)	X	X
QC for Sequencing Reads		X
QC for Read Mapping		X
QC for Targeted Sequencing		X
Quality-based Variant Detection (legacy)	X	X
Remove False Positives		X
Remove Germline Variants		X
Remove Reference Variants		X
Remove Variants Found in Common dbSNP		X
Remove Variants Found in External Database		X
Remove Variants Found in HapMap		X
Remove Variants Found in 1000 Genomes Project		X
Remove Variants Inside Genome Regions		X
Remove Variants Not Found in External Database		X
Remove Variants Outside Genome Regions		X
Remove Variants Outside Targeted Regions		X
RNA-Seq Analysis	X	X
Screen Ligands		
Trim Sequences	X	X
Trio Analysis	X	X

Please note that a static license has a limitation on the maximum number of cores, see section 1.3.1.

Appendix B

Reference data overview

Human hg19

- **Human reference sequence, ENSEMBL**
  ftp://ftp.ensembl.org/pub/current_fasta/homo_sapiens/dna/
  Chromosomes 1-22, X, Y and M human reference DNA sequence GRCh37(hg19)

- **Human genes, coding sequences and transcripts, ENSEMBL**
  ftp://ftp.ensembl.org/pub/current_gtf/homo_sapiens/
  All annotated protein coding genes for human reference sequence GRCh37(hg19). The annotation was done by ENSEMBL and includes annotations from RefSeq, CCDS as well as ENSEMBL itself.

- **HapMap variants, ENSEMBL**
  ftp://ftp.ensembl.org/pub/current_variation/gvf/homo_sapiens/
  The goal of the International HapMap Project is to develop a haplotype map of the human genome, the HapMap, which will describe the common patterns of human DNA sequence variation (for more information about HapMap see http://hapmap.ncbi.nlm.nih.gov/). It is recommended that you configure your workflows with the file from the population that best matches the ethnicity of the patient from which the sample was taken. You can find more about the population codes, which are part of the filename here: http://www.sanger.ac.uk/resources/downloads/human/hapmap3.html

- **Variants found by the 1000 Genomes Project, ENSEMBL**
  ftp://ftp.ensembl.org/pub/current_variation/gvf/homo_sapiens/
  The 1000 Genomes Project Phase 1 created an integrated map of genetic variations from 1092 human genomes [Consortium et al., 2012]. It is recommended that you configure your workflows with the file from the population that bests matches the ethnicity of patient from which the sample was taken. You can learn more about the population codes that are part of the filename here: http://www.1000genomes.org/.

- **dbSNP variants, UCSC**
  http://hgdownload.soe.ucsc.edu/goldenPath/hg19/database/snp138.txt.gz
  Human variants present in the Single Nucleotide Polymorphism Database (dbSNP), which includes smaller insertions, deletions, replacements, SNPs and MNVs. Please note that most

variants in dbSNP are not validated and everybody can submit data to dbSNP. The collection of variants includes clinical relevant as well as common variants. Please note that the url must be modified according to what you would like to download - e.g. if you are interested in `snp141Common.txt.gz`, "138" in the url should be replaced with "141Common" (for a full list see http://hgdownload.soe.ucsc.edu/goldenPath/hg19/database/).

- **dbSNP common variants, UCSC**
  http://hgdownload.soe.ucsc.edu/goldenPath/hg19/database/snp138Common.txt.gz
  Uniquely mapped variants that appear in at least 1% of the population or are 100% non-reference. Please note that the url must be modified according to what you would like to download - e.g. if you are interested in `snp141Common.txt.gz`, "138" in the url should be replaced with "141" (for a full list see http://hgdownload.soe.ucsc.edu/goldenPath/hg19/database/)

- **ClinVar database variants, NCBI**
  http://www.ncbi.nlm.nih.gov/clinvar/docs/maintenance_use/
  ClinVar is designed to provide a freely accessible, public archive of reports of the relationships among human variations and phenotypes, with supporting evidence.

- **PhastCons Conservation Scores, UCSC**
  http://hgdownload.soe.ucsc.edu/goldenPath/hg19/phastCons100way/hg19.100way.phastCons/
  Conservation track of UCSC from a multiple alignments of 100 species and measurements of evolutionary conservation using the phastCons algorithm from the PHAST package.

- **Human Gene Ontology (GO slim) file, EBI**
  http://www.ebi.ac.uk/QuickGO/GMultiTerm
  Gene Ontology file in slim format (only high level GO terms annotated) for the GO categories Molecular Function, Biological Process and Cellular Component annotated on human genes. The file was made using the QuickGO tool from the EBI (http://www.ebi.ac.uk/QuickGO/ GMultiTerm).

- **target primers and target regions QIAGEN_v2**
  https://www.qiagen.com/dk/shop/sample-technologies/dna-sample-technologies/genomic-dna/generead-dnaseq-gene-panels-v2/
  These primers and regions are defined and provided for by QIAGEN GeneRead DNAseq Targeted Panels V2.

Human hg38

- **Human reference sequence, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/fasta/homo_sapiens/dna/
  The file `Homo_sapiens.GRCh38.dna.toplevel.fa.gz` has chromosomal sequences along with several scaffolds. The scaffolds were removed in the workbench.

- **Human genes, coding sequences and transcripts, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/gtf/homo_sapiens/
  filename: `Homo_sapiens.GRCh38.80.gtf.gz`

- **HapMap variants, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/variation/gvf/homo_sapiens/
  The goal of the International HapMap Project is to develop a haplotype map of the human genome, the HapMap, which will describe the common patterns of human DNA sequence variation (for more information about HapMap see http://hapmap.ncbi.nlm.nih.gov/). It is recommended that you configure your workflows with the file from the population that best matches the ethnicity of the patient from which the sample was taken. You can find more about the population codes, which are part of the filename here: http://www.sanger.ac.uk/resources/downloads/human/hapmap3.html

- **Variants found by the 1000 Genomes Project, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/variation/gvf/homo_sapiens/
  The 1000 Genomes Project Phase 1 created an integrated map of genetic variations from 1092 human genomes [Consortium et al., 2012]. It is recommended that you configure your workflows with the file from the population that bests matches the ethnicity of patient from which the sample was taken. You can learn more about the population codes that are part of the filename here: http://www.1000genomes.org/.

- **dbSNP variants, UCSC**
  http://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/
  Human variants present in the Single Nucleotide Polymorphism Database (dbSNP), which includes smaller insertions, deletions, replacements, SNPs and MNVs. Please note that most variants in dbSNP are not validated and everybody can submit data to dbSNP. filename: `snp142.txt.gz`

- **dbSNP common variants, UCSC**
  http://hgdownload.soe.ucsc.edu/goldenPath/hg38/database/
  Uniquely mapped variants that appear in at least 1% of the population or are 100% non-reference. filename: `snp142Common.txt.gz`

- **ClinVar database variants, NCBI**
  ftp://ftp.ncbi.nlm.nih.gov/pub/clinvar/vcf_GRCh38/
  ClinVar is designed to provide a freely accessible, public archive of reports of the relationships among human variations and phenotypes, with supporting evidence. filename: `clinvar_20150629.vcf`

- **PhastCons Conservation Scores, UCSC**
  http://hgdownload.cse.ucsc.edu/goldenPath/hg38/phastCons20way/
  Conservation track of UCSC from a multiple alignments of 100 species and measurements of evolutionary conservation using the phastCons algorithm from the PHAST package. filename: `hg38.phastCons20way.wigFix`

- **Human Gene Ontology (GO slim) file, EBI**
  http://www.ebi.ac.uk/QuickGO/GMultiTerm
  Gene Ontology file in slim format (only high level GO terms annotated) for the GO categories Molecular Function, Biological Process and Cellular Component annotated on human genes. The file was made using the QuickGO tool from the EBI (http://www.ebi.ac.uk/QuickGO/ GMultiTerm).

- **target primers and target regions QIAGEN_v2**
  https://www.qiagen.com/dk/shop/sample-technologies/dna-sample-technologies/

genomic-dna/generead-dnaseq-gene-panels-v2/
These primers and regions are defined and provided for by QIAGEN GeneRead DNAseq Targeted Panels V2.

Mouse Mm10

- **Mouse reference sequence, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/fasta/mus_musculus/dna/
  The file `Mus_musculus.GRCm38.dna_sm.toplevel.fa.gz` has chromosomal sequences along with several scaffolds. The scaffolds were removed in the workbench.
- **Mouse genes, coding sequences and transcripts, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/gtf/mus_musculus/
  filename: `Mus_musculus.GRCm38.80.gtf.gz`
- **dbSNP variants, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-80/variation/gvf/mus_musculus/
  filename: `Mus_musculus.gvf.gz`
- **PhastCons Conservation Scores, UCSC**
  http://hgdownload.cse.ucsc.edu/goldenPath/mm10/phastCons60way/mm10.60way.phastCons/
  Each chromosome has a separate wigfix file. Each needs to be downloaded (22 files) and then combined to make single wigfix file before importing in workbench. filename: `*.phastCons60way.wigFix.gz`
- **Mouse Gene Ontology (GO slim) file, EBI**
  http://www.ebi.ac.uk/QuickGO/GMultiTerm
  Gene Ontology file in slim format (only high level GO terms annotated) for the GO categories Molecular Function, Biological Process and Cellular Component annotated on mouse genes. The file was made using the QuickGO tool from the EBI (http://www.ebi.ac.uk/QuickGO/ GMultiTerm).

Rat Rnor5.0

- **Rat reference sequence, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-79/fasta/rattus_norvegicus/dna/
  The file `Rattus_norvegicus.Rnor_5.0.dna.toplevel.fa.gz` has chromosomal sequences along with several scaffolds. The scaffolds were removed in the workbench.
- **Rat genes, coding sequences and transcripts, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-79/gtf/rattus_norvegicus
  filename: `Rattus_norvegicus.Rnor_5.0.79.gtf.gz`
- **dbSNP variants, ENSEMBL**
  ftp://ftp.ensembl.org/pub/release-79/variation/gvf/rattus_norvegicus/
  filename: `Rattus_norvegicus.gvf.gz`

- **PhastCons Conservation Scores, UCSC**
  http://hgdownload.cse.ucsc.edu/goldenPath/rn5/phastCons13way/
  Each chromosome has a separate wigfix file. Each needs to be downloaded (22 files) and then combined to make single wigfix file before importing in workbench. filename: `phastCons13way.wigFix.gz`

- **Rat Gene Ontology (GO slim) file, EBI**
  http://www.ebi.ac.uk/QuickGO/GMultiTerm
  Gene Ontology file in slim format (only high level GO terms annotated) for the GO categories Molecular Function, Biological Process and Cellular Component annotated on mouse genes. The file was made using the QuickGO tool from the EBI (http://www.ebi.ac.uk/QuickGO/ GMultiTerm).

Appendix C

Proteolytic cleavage enzymes

Most proteolytic enzymes cleave at distinct patterns. Below is a compiled list of proteolytic enzymes used in *Biomedical Genomics Workbench*.

Name	P4	P3	P2	P1	P1’	P2’
Cyanogen bromide (CNBr)	-	-	-	M	-	-
Asp-N endopeptidase	-	-	-	-	D	-
Arg-C	-	-	-	R	-	-
Lys-C	-	-	-	K	-	-
Trypsin	-	-	-	K, R	not P	-
Trypsin	-	-	W	K	P	-
Trypsin	-	-	M	R	P	-
Trypsin*	-	-	C, D	K	D	-
Trypsin*	-	-	C	K	H, Y	-
Trypsin*	-	-	C	R	K	-
Trypsin*	-	-	R	R	H,R	-
Chymotrypsin-high spec.	-	-	-	F, Y	not P	-
Chymotrypsin-high spec.	-	-	-	W	not M, P	-
Chymotrypsin-low spec.	-	-	-	F, L, Y	not P	-
Chymotrypsin-low spec.	-	-	-	W	not M, P	-
Chymotrypsin-low spec.	-	-	-	M	not P, Y	-
Chymotrypsin-low spec.	-	-	-	H	not D, M, P, W	-
o-Iodosobenzoate	-	-	-	W	-	-
Thermolysin	-	-	-	not D, E	A, F, I, L, M or V	-
Post-Pro	-	-	H, K, R	P	not P	-
Glu-C	-	-	-	E	-	-
Asp-N	-	-	-	-	D	-
Proteinase K	-	-	-	A, E, F, I, L, T, V, W, Y	-	-
Factor Xa	A, F, G, I, L, T, V, M	D,E	G	R	-	-
Granzyme B	I	E	P	D	-	-
Thrombin	-	-	G	R	G	-
Thrombin	A, F, G, I, L, T, V, M	A, F, G, I, L, T, V, W, A	P	R	not D, E	not D, E
TEV (Tobacco Etch Virus)	-	Y	-	Q	G, S	-

Appendix D

Restriction enzymes database configuration

Biomedical Genomics Workbench uses enzymes from the **REBASE** restriction enzyme database at http://rebase.neb.com. If you wish to add enzymes to this list, you can do this by manually using the procedure described here.

Note! Please be aware that this process needs to be handled carefully, otherwise you may have to re-install the Workbench to get it to work.

First, download the following file: http://www.resources.qiagenbioinformatics.com/wbsettings/link_emboss_e_custom. In the Workbench installation folder under `settings`, create a folder named `rebase` and place the extracted `link_emboss_e_custom` file here.

Note that in macOS, the extension file "link_emboss_e_custom" will have a ".txt" extension in its filename and metadata that needs to be removed. Right click the file name, choose "Get info" and remove ".txt" from the "Name & extension" field.

Open the file in a text editor. The top of the file contains information about the format, and at the bottom there are two example enzymes that you should replace with your own.

Please note that the CLC Workbenches only support the addition of 2-cutter enzymes. Further details about how to format your entries accordingly are given within the file mentioned above.

After adding the above file, or making changes to it, you must restart the Workbench for changes take effect.

Appendix E

Technical information about modifying Gateway cloning sites

The *Biomedical Genomics Workbench* comes with a pre-defined list of Gateway recombination sites. These sites and the recombination logics can be modified by downloading and editing a properties file. Note that this is a technical procedure only needed if the built-in functionality is not sufficient for your needs.

The properties file can be downloaded from `http://www.resources.qiagenbioinformatics.com/wbsettings/gatewaycloning.zip`. Extract the file included in the zip archive and save it in the `settings` folder of the Workbench installation folder. The file you download contains the standard configuration. You should thus update the file to match your specific needs. See the comments in the file for more information.

The name of the properties file you download is `gatewaycloning.1.properties`. You can add several files with different configurations by giving them a different number, e.g. `gatewaycloning.2.properties` and so forth. When using the Gateway tools in the Workbench, you will be asked which configuration you want to use (see figure E.1).

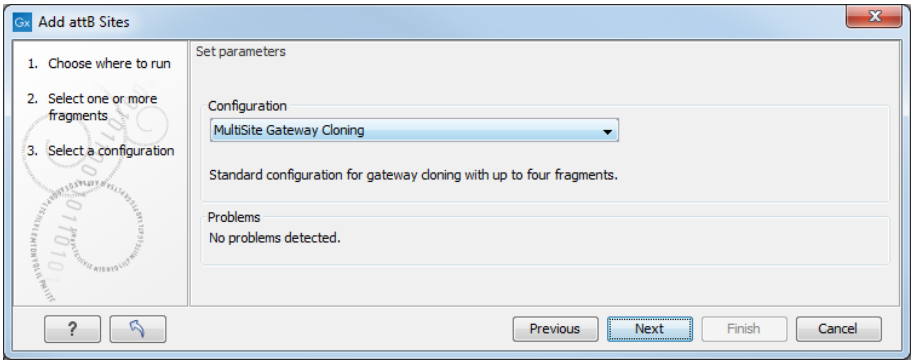


Figure E.1: *Selecting between different gateway cloning configurations.*

Appendix F

IUPAC codes for amino acids

(Single-letter codes based on International Union of Pure and Applied Chemistry)

The information is gathered from: http://www.insdc.org/documents/feature_table.html

One-letter abbreviation	Three-letter abbreviation	Description
A	Ala	Alanine
R	Arg	Arginine
N	Asn	Asparagine
D	Asp	Aspartic acid
C	Cys	Cysteine
Q	Gln	Glutamine
E	Glu	Glutamic acid
G	Gly	Glycine
H	His	Histidine
J	Xle	Leucine or Isoleucineucine
L	Leu	Leucine
I	ILe	Isoleucine
K	Lys	Lysine
M	Met	Methionine
F	Phe	Phenylalanine
P	Pro	Proline
O	Pyl	Pyrrolysine
U	Sec	Selenocysteine
S	Ser	Serine
T	Thr	Threonine
W	Trp	Tryptophan
Y	Tyr	Tyrosine
V	Val	Valine
B	Asx	Aspartic acid or Asparagine Asparagine
Z	Glx	Glutamic acid or Glutamine Glutamine
X	Xaa	Any amino acid

Appendix G

IUPAC codes for nucleotides

(Single-letter codes based on International Union of Pure and Applied Chemistry)

The information is gathered from: http://www.iupac.org and http://www.insdc.org/documents/feature_table.html.

Code	Description
A	Adenine
C	Cytosine
G	Guanine
T	Thymine
U	Uracil
R	Purine (A or G)
Y	Pyrimidine (C, T, or U)
M	C or A
K	T, U, or G
W	T, U, or A
S	C or G
B	C, T, U, or G (not A)
D	A, T, U, or G (not C)
H	A, T, U, or C (not G)
V	A, C, or G (not T, not U)
N	Any base (A, C, G, T, or U)

Appendix H

Formats for import and export

H.1 List of bioinformatic data formats

Below is a list of bioinformatic data formats, i.e. formats for importing and exporting molecule structures, sequences, alignments and trees.

H.1.1 Molecule structure formats

File type	Suffix	Import	Export	Description
PDB	.pdb	X	X	
Tripos Mol2	.mol2	X	X	
MDL Mol	.sdf	X		
CLC	.clc	X	X	Rich format including all information

H.1.2 Sequence data formats

File type	Suffix	Import	Export	Description
AB1	.ab1	X		Including chromatograms
ABI	.abi	X		Including chromatograms
CLC	.clc	X	X	Rich format including all information
Clone manager	.cm5	X		Clone manager sequence format
FASTA	.fsa/.fasta	X	X	Simple format, name & description
Raw sequence	any	X		Only sequence (no name)
Sequence CSV	.csv	X	X	Simple format. One seq per line: name, description(optional), sequence
Tab delimited text	.txt		X	Annotations in tab delimited text format
Phred	.phd	X		Including chromatograms
PIR(NBRF)	.pir	X	X	Simple format, name and description
SCF2	.scf	X		Including chromatograms
SCF3	.scf	X	X	Including chromatograms
Staden	.sdn	X		
Swiss-Prot	.swp	X		Rich information incl. annotations (only peptides)

Note that high-throughput sequencing data formats from Illumina, SOLiD, IonTorrent, 454 and also high-throughput fasta and trace files are imported using a special import as described in section 6.3. These data can also be exported in fastq format (using NCBI/Sanger Phred quality scores).

When exporting in fasta format, it is possible to remove sequence ends covered by annotations of type "Trim" (read more in section 33.2).

H.1.3 Read mapping formats

File type	Suffix	Import	Export	Description
ACE	.ace	X	X	No chromatogram or quality score
AGP	.agp/.fa		X	Exports scaffolded contigs (see below)
BAM	.bam	X	X	Compressed version of SAM. See details in section 6.3.8
CLC	.clc	X	X	Rich format including all information
CLC Assembly File	.cas	X		Output from the CLC Assembly Cell
SAM	.sam	X	X	Sequence Alignment/Map. See details in section 6.3.8
Mapping coverage	.tsv		X	Detailed per-base info on coverage (see below)

Note about BAM export Index files can be created as part of BAM exports.

Note about AGP format Both sequence lists and contigs with reads mapped can be used. Based on annotations of type **Scaffold** (which are automatically added when running the *de novo* assembly with the scaffold option), the contigs are broken up before exported as fasta. The agp

file produced holds information about how the contigs relate to each other.

Export of coverage information from sequence alignments Coverage information from read mappings can be exported in a tabular format using the **Mapping coverage** export. The output contains information on the number of nucleotides aligned to positions in reference sequences. Insertions are also reported as described below while deletions are reported as reference regions without read coverage. Both stand-alone read mappings and read tracks can be used as input.

The exported file contains the following columns:

Column	Description
1	Reference name
2	Reference position
3	Reference sub-position (insertion)
4	Reference symbol
5	Number of A's
6	Number of C's
7	Number of G's
8	Number of T's
9	Number of N's
10	Number of Gaps
11	Total number of reads covering the position

The **Reference sub-position** column is empty (indicated by a - symbol) when the reference is defined at a given position. In case of an insertion this column contains an index into the insertion (a number between 1 and the length of the insertion) while the **Reference symbol** column is empty and the **Reference position** column contains the position of the last reference.

H.1.4 Contig formats

File type	Suffix	Import	Export	Description
ACE	.ace	X	X	No chromatogram or quality score
CLC	.clc	X	X	Rich format including all information

H.1.5 Alignment formats

File type	Suffix	Import	Export	Description
Aligned fasta	.fa	X	X	Simple fasta-based format with – for gaps
CLC	.clc	X	X	Rich format including all information
ClustalW	.aln	X	X	
GCG Alignment	.msf	X	X	
Nexus	.nxs/.nexus	X	X	
Phylip Alignment	.phy	X	X	

H.1.6 Expression data formats

Read about technical details of these data formats in section J.

File type	Suffix	Import	Export	Description
Affymetrix CHP	.chp/.psi	X		Expression values and annotations
Affymetrix pivot/metric	.txt/.csv	X		Gene-level expression values
Affymetrix NetAffx	.csv	X		Annotations
CLC	.clc	X	X	Rich format including all information
Excel	.xls/.xlsx		X	All tables and reports
Generic	.txt/.csv	X		Expression values
Generic	.txt/.csv	X		Annotations
GEO soft sample/series	.txt/.csv	X		Expression values
Illumina	.txt	X		Expression values and annotations
Table CSV	.csv		X	Samples and experiments
Tab delimited	.txt		X	Samples and experiments

H.1.7 Annotation and variant formats

Please note that all of the annotation and variant formats can be imported as tracks (see section 6.2). GFF, GVF and GTF formats can also be imported as annotations on a standard (i.e., non-track) sequence or sequence list using functionality provided by the Annotate with GFF plugin (http://www.qiagenbioinformatics.com/plugins/annotate-with-gff-file/).

File type	Suffix	Import	Export	Description
Annotation CSV export	.csv		X	Annotations in csv format
Annotation Excel 2010	.xlsx		X	Annotations in Excel format
Annotation Excel 97 - 2007	.xls		X	Annotations in Excel format
VCF	.vcf	X	X	See note below
GFF	.gff	X		To import as annotation track, see section 6.2.
GVF	.gvf	X	X	Special version of GFF for variant data. See GFF entry above.
GTF	.gtf	X	X	Special version of GFF for gene annotation data. See GFF entry above.
GFF3	.gff3	X	X	To import and export as annotation track, see section 6.2.1.
COSMIC variation database	.tsv	X		Special format for COSMIC data
BED	.bed	X	X	See section 6.2
Wiggle	.wig	X	X	See section 6.2
UCSC variant database table dump	.txt	X		See section 6.2
Complete genomics master var files	masterVar	X		Complete genomics variant data format

Special note on VCF export

For VCF export, counts from the variant track are put in CLCAD2 tags and coverage in DP tags. The values of the CLCAD2 tag follow the order of REF and ALT, with one value for the REF and for each ALT. For example if there has been a homozygote variant identified at a certain position, the value of the GT field is 1/1 and the corresponding CLCAD2 value for the reference allele will be 0, which is always the first number in the CLCAD2 field. Please note that this does not mean the original *mapping* did not have any reads with that sequence, but it means that the *variant track* being exported does not contain the reference allele.

When exporting VCF files, there are three options:

Reference sequence track Since the VCF format specifies that reference and allele sequences cannot be empty, deletions and insertions have to be padded with bases from the reference sequence. The export needs access to the reference sequence track in order to find the neighboring bases.

Enforce diploid export The *Biomedical Genomics Workbench* option will generate a VCF file in which the allele values in the Genotype (GT) field for haploid variants are reported following

the format for diploid variants (i.e. the GT allele values reported are 1/1). This is to ensure compatibility of the exported VCF file with programs for downstream variant analysis that expect strictly diploid genomes. The user can specify that the Enforce diploid option is only applied to certain chromosomes, while others may be reported as haploid. If you export a variant track that has been filtered, there can be situations where there is only one heterozygous variant at a given position. In this case, the *Biomedical Genomics Workbench* will use a "." to denote an unknown genotype, so the GT field will be "1/.".

Note: the "Enforce diploid" option does NOT enforce diploidy for polyploid variant loci. Regardless of this setting, all variant alleles reported during variant calling are included in the exported VCF file.

It is important to note that this **Enforce diploid export** option will create a diploid format of the VCF file, but it is not able to recover any inconsistencies in the variant track used as input. If the variant track has three variants at a given position, three genotypes will be output. Or if the variant track has two variants at the same position that both postulate to be homozygous, they will be output as two heterozygous variants. When exporting data created by the variant callers of *Biomedical Genomics Workbench*, this is usually not a problem, but when applying this diploid scheme to data that has been imported into the *Biomedical Genomics Workbench* from other sources, the data can be inconsistent with a diploid model.

Exceptions Some chromosomes can be excepted from the enforced diploid export. For a human genome, that would be relevant for the mitochondrion and for male X and Y chromosomes. For this option, you can select which chromosomes should be excepted. They will be exported in the standard way without assuming there should be two genotypes, and homozygous calls will just have one value in the GT field.

Special note on former VCF export

In CLC Genomics Workbench 6.5 instead of the CLCAD2, the CLCAD field had been reported. The difference between CLCAD and CLCAD2 is that the former is following the order in the GT (genotype) field in VCF, while the latter is following the order of the REF and ALT fields in VCF in is therefore more in line with the AD field reported from GATK and other sources.

Special notes on VCF import Note! Please also see section 6.2.

The import process for VCF files into the CLC Genomics Workbench currently work as follows:

1. For VCF rows that are reporting the reference base, no variants are imported.
2. In cases where GT = 0/0 or GT = ./., no variants are imported at all.
3. In cases where GT = X/. or GT = ./X , and where X is not zero, a single variant is imported depending on the actual value of X.
4. In cases where GT = X/X and X is not zero, in Genomics Workbench 6.5 this will result in two independent variants. In version 6.5.1 they will be reported as a single homozygous variant.
5. In cases where GT = X/Y and X and Y are different but either one may be zero, two independent variants are created.

Special notes on chromosome names synonyms used during import

When importing annotations as tracks, we try to make things simple for the user by having a set of chromosome names that are recognized as synonyms. The check on the chromosome name comparison is made by looking through the chromosomes in the order in which they are registered in the genome. The first match with any of the synonym names for a given chromosome is the chromosome to which the information will be added.

The synonyms applied are:

For any number N between (including) 1 and 22:

N, chrN, chromosome_N, and NC_00000N are seen as meaning the same thing. As concrete examples:

1 == chr1 == chromosome_1 == NC_000001

22 == chr22 == chromosome_22 == NC_000022

For any number N larger than 23:

N, chrN, chromosome_N are seen as meaning the same thing. As a concrete example:

26 == chr26 == chromsome_26

For chromsome names with letters, not numbers:

X, chrX, and chromosome_X and NC_000023 are synonyms.

Y, chrY, chromosome_Y and NC_000024 are synonyms.

M, MT, chrM, chrMT, chromosome_M, chromosome_MT and NC_001807 are synonyms.

The accession numbers in the listings above (NC_XXXXXX) allow for the matching against NCBI hg19 human reference names against the names used by USCS and vitally, the names used by Ensembl. Thus, in this case, if you have the correct number of chromosomes in a human reference (i.e. 25 references, including the hg19 mitochondria), that set of tracks can be used as the basis for downloading/importing annotations via Download Genomes, for example.

Note: These rules only apply for importing annotations as tracks, whether that is directly or via Download Genomes. Synonyms are not applied when doing BAM imports or when using the Annotate with GFF plugin. There, your reference names in the Workbench must exactly match the references names used in your BAM file or GFF/GTF/GVF file respectively.

H.1.8 Table and text formats

File type	Suffix	Import	Export	Description
Excel	.xls/.xlsx	X	X	All tables and reports
Table CSV	.csv	X	X	All tables
Tab delimited	.txt		X	All tables
Text	.txt	X	X	All data in a textual format
CLC	.clc	X	X	Rich format including all information
HTML	.html		X	All tables
PDF	.pdf		X	Export reports in Portable Document Format

Please see table H.1.6 **Expression data formats** for special cases of table imports.

H.1.9 File compression formats

File type	Suffix	Import	Export	Description
Zip export	.zip		X	Selected files in CLC format
Zip import	.zip/.gz/.tar	X		Contained files/folder structure (.tar and .zip not supported for NGS data)

Note! It is possible to import 'external' files into the Workbench and view these in the **Navigation Area**, but it is only the above mentioned formats whose *contents* can be shown in the Workbench.

H.2 List of graphics data formats

Below is a list of formats for exporting graphics. All data displayed in a graphical format can be exported using these formats. Data represented in lists and tables can only be exported in .pdf format (see section 6.7 for further details).

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

Appendix I

SAM/BAM export format specification

SAM Specification The workbench aims to import and export SAM and BAM files according to the v1.4-r962 version of the SAM specification (see http://samtools.sourceforge.net/SAM1.pdf). This appendix describes how the workbench exports SAM and BAM files along with known limitations.

SAM and BAM Export - General notes

The SAM exporter writes unsorted SAM and BAM files.

If the reference name contains spaces, the spaces are removed. Each occurrence of '=' (equals sign) and '@' (at sign) in a reference name is replaced by an '_' (underscore).

The SAM importer and exporter support the ID, SM, PI and PL read group tags. All other read group tags are ignored.

The BAM exporter can also output additional annotations added by tools provided by plugins, and where that is the case, further details are provided in the plugin manual.

SAM Alignment Section A few remarks on the exported alignment section:

- Unmapped reads are not exported.
- If pairs are not on the same contig, the mates will be exported as single reads.
- Multi segment mappings will be imported as a paired data set.
- If a read name contains spaces, the spaces are replaced by an underscore '_'.
- The exported CIGAR string uses 'M' to indicate match or mismatch and does not use '=' (equals sign) or 'X'.
- CLC software does not support or record mapping quality for read mappings. To fulfill the requirement in the BAM format specifications that a read mapping quality is recorded for all mapped reads, the values 0 and 60 are used when mappings are exported from the Workbench. The value 60 is given to reads that mapped uniquely. The value 0 is given to reads that could map equally well to other locations besides the one being reported in the BAM file.

  This scoring system is based on a recommendation provided in the SAM FAQ:

http://sourceforge.net/apps/mediawiki/samtools/index.php?title=SAM_FAQ#How_to_make_my_aligner_work_best_with_samtools.3F

Optional fields in the alignment section The following is true for the export of optional fields:

- The NH tag is exported.
- The NM tag is not exported.
- The workbench exports color space information in the CS tag.
- The colors of a right mate are incorrect since the colors of a paired read are stored as a single color string.
- For hard clipped sequence reads, the color space is incorrect, since the color space string is not hard clipped.
- SAM files contain sequence quality score and color quality scores. The workbench only have color quality scores and these are stored and exported as sequence quality scores.

I.1 Flags

The workbench's use of the alignment flags is shown in the following table and subsequent examples.

Bit	SAM description	Usage in Workbench
0x1	template having multiple segments in sequencing	set if the segment is part of a pair
0x2	each segment properly aligned according to the aligner	set if the pair is not broken
0x4	segment unmapped	never set since the exporter does not export unmapped reads
0x8	next segment in the template unmapped	never set by the exporter. If a segment has an unmapped mate, the flag 0x1 is not set for the segment, i.e. it is not output as part of a pair
0x10	SEQ being reverse complemented	set if and only if the segment was reverse complemented during mapping
0x20	SEQ of the next segment in the template being reversed	set if and only if the mate was reverse complemented during mapping
0x40	the first segment in the template	this mate is the first segment of the pair
0x80	the last segment in the template	this mate is the second segment of the pair
0x100	secondary alignment	never set by the exporter. No reads with this flag set are imported[1].
0x200	not passing quality controls	never set by the exporter and ignored by the importer
0x400	PCR or optical duplicate	never set by the exporter and ignored by the importer

Flag Examples

[1]The representation of a particular read with more than one location in a mapping is not supported in the software and thus cannot be imported.

The following table illustrates some of the possible flags in the workbench.

Description of the example	Bits	Flag	Illustration
The first mate of a non-broken paired read	`0x1`, `0x2`, `0x20`, `0x40`	**99**	See Figure I.1
The second mate of a non-broken paired read	`0x1`, `0x2`, `0x10`, `0x80`	**147**	See Figure I.2
A single, forward read (or paired read, where only one mate of the pair is mapped)	No set bits	**0**	see Figure I.3
A single, reversed read (or paired read, where only one mate of the pair is mapped)	`0x10`	**16**	See Figure I.4
The first, forward segment from a broken pair with forward mate	`0x1`, `0x40`	**65**	See Figure I.5
The second, forward segment from broken pair with reversed mate	`0x1`, `0x20`, `0x80`	**161**	See Figure I.6
The first, reversed segment from broken pair with forward mate	`0x1`, `0x10`, `0x40`	**81**	See Figure I.7
The second, reversed segment from broken pair with reversed mate	`0x1`, `0x10`, `0x20`, `0x80`	**177**	See Figure I.8

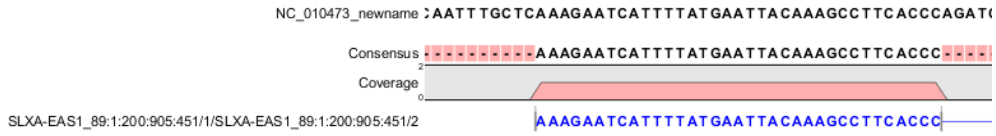


Figure I.1: *The read is paired, both reads are mapped and the mate of this read is reversed*

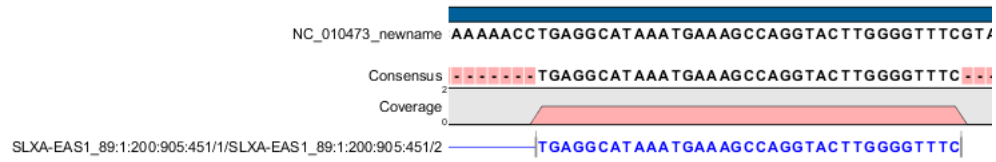


Figure I.2: *The read is paired, both mates are mapped, and this segment is reversed*

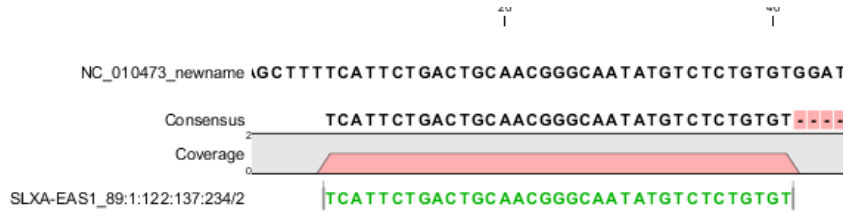


Figure I.3: *A single, forward read, or a paired read where the mate is not mapped*

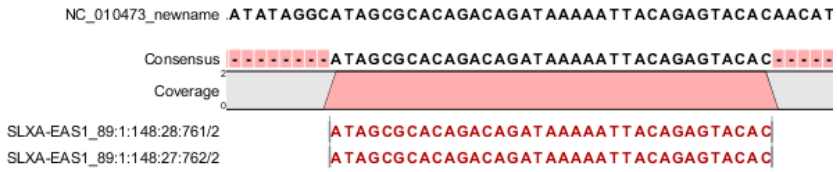


Figure I.4: *The read is a single, reversed read, or a paired read where the mate is not mapped*

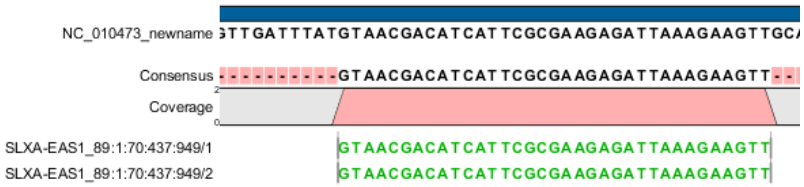


Figure I.5: *These forward reads are paired. They map to the same place, so the pair is broken*

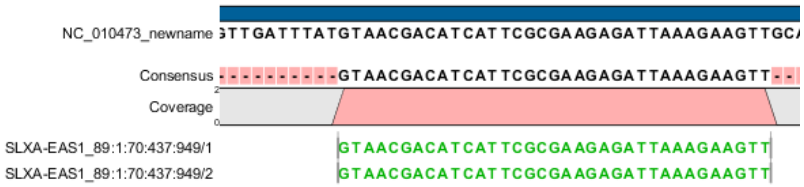


Figure I.6: *Forward read that is part of a broken read where the mate is reversed*

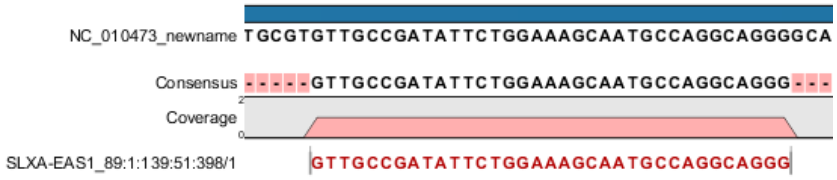


Figure I.7: *Reversed read that is part of a broken pair, where the mate is forward*

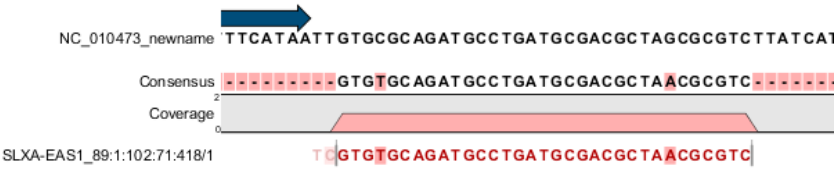


Figure I.8: *Reversed read that is part of a broken pair, where the mate is also reversed.*

Appendix J

Gene expression annotation files and microarray data formats

The workbench supports analysis of one-color expression arrays. These may be imported from GEO soft sample- or series- file formats, or for Affymetrix arrays, tab-delimited pivot or metrics files, or from Illumina expression files. Expression array data from other platforms may be imported from tab, semi-colon or comma separated files containing the expression feature IDs and levels in a tabular format (see section J.5).

The workbench assumes that expression values are given at the gene level, thus probe-level analysis of e.g. Affymetrix GeneChips and import of Affymetrix CEL and CDF files is currently not supported. However, the workbench allows import of txt files exported from R containing processed Affymetrix CEL-file data (see see section J.2).

Affymetrix NetAffx annotation files for expression GeneChips in csv format and Illumina annotation files can also be imported. Also, you may import your own annotation data in tabular format see section J.5).

Below you find descriptions of the microarray data formats that are supported by *Biomedical Genomics Workbench*. Note that we for some platforms support both expression data and annotation data.

J.1 GEO (Gene Expression Omnibus)

The GEO (Gene Expression Omnibus) sample and series formats are supported. Figure J.1 shows how to download the data from GEO in the right format. GEO is located at `http://www.ncbi.nlm.nih.gov/geo/`.

The GEO sample files are tab-delimited .txt files. They have three required lines:

```
^SAMPLE = GSM21610
!sample_table_begin
...
!sample_table_end
```

The first line should start with `^SAMPLE =` followed by the sample name, the line `!sample_table_begin`

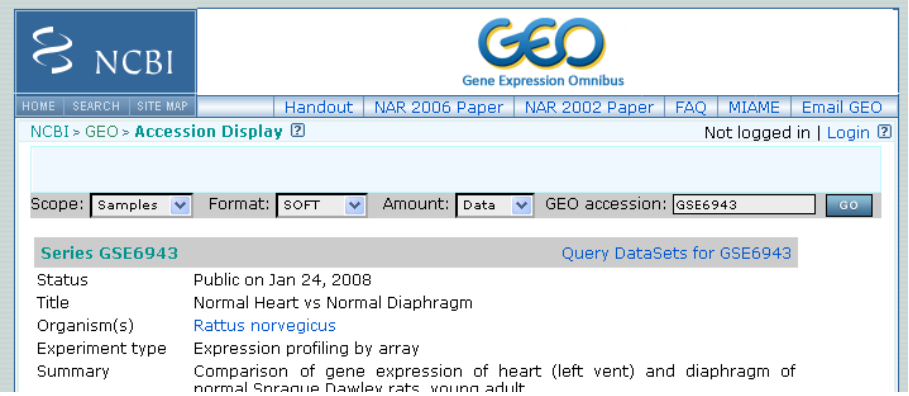


Figure J.1: *Selecting Samples, SOFT and Data before clicking go will give you the format supported by the* **Biomedical Genomics Workbench**.

and the line `!sample_table_end`. Between the `!sample_table_begin` and `!sample_table_end`, lines are the column contents of the sample.

Note that GEO sample importer will also work for concatenated GEO sample files — allowing multiple samples to be imported in one go. Download a sample file containing concatenated sample files here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFilesConcatenated.txt

Below you can find examples of the formatting of the GEO formats.

J.1.1 GEO sample file, simple

This format is very simple and includes two columns: one for feature id (e.g. gene name) and one for the expression value.

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF    VALUE
id1       105.8
id2       32
id3       50.4
id4       57.8
id5       2914.1
!sample_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimple.txt

J.1.2 GEO sample file, including present/absent calls

This format includes an extra column for absent/present calls that can also be imported.

```
^SAMPLE = GSM21610
```

```
!sample_table_begin
ID_REF    VALUE    ABS_CALL
id1       105.8    M
id2       32       A
id3       50.4     A
id4       57.8     A
id5       2914.1   P
!sample_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresent.txt

J.1.3 GEO sample file, including present/absent calls and p-values

This format includes two extra columns: one for absent/present calls and one for absent/present call p-values, that can also be imported.

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF     VALUE      ABS_CALL      DETECTION P-VALUE
id1        105.8      M             0.00227496
id2        32         A             0.354441
id3        50.4       A             0.904352
id4        57.8       A             0.937071
id5        2914.1     P             6.02111e-05
!sample_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFileAbsentPresentC
txt

J.1.4 GEO sample file: using absent/present call and p-value columns for sequence information

The workbench assumes that if there is a third column in the GEO sample file then it contains present/absent calls and that if there is a fourth column then it contains p-values for these calls. This means that the contents of the third column is assumed to be text and that of the fourth column a number. As long as these two basic requirements are met, the sample should be recognized and interpreted correctly.

You can thus use these two columns to carry additional information on your probes. The absent/present column can be used to carry additional information like e.g. sequence tags as shown below:

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF       VALUE      ABS_CALL
```

```
id1         105.8       AAA
id2         32          AAC
id3         50.4        ATA
id4         57.8        ATT
id5         2914.1      TTA
!sample_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequence
txt

Or, if you have multiple probes per sequence you could use the present/absent column to hold the sequence name and the p-value column to hold the interrogation position of your probes:

```
^SAMPLE = GSM21610
!sample_table_begin
ID_REF    VALUE    ABS_CALL    DETECTION P-VALUE
probe1    755.07   seq1        1452
probe2    587.88   seq1        497
probe3    716.29   seq1        1447
probe4    1287.18  seq2        1899
!sample_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSampleFileSimpleSequence
txt

J.1.5 GEO series file, simple

The series file includes expression values for multiple samples. Each of the samples in the file will be represented by its own element with the sample name. The first row lists the sample names.

```
!Series_title "Myb specificity determinants"
!series_matrix_table_begin
"ID_REF" "GSM21610" "GSM21611" "GSM21612"
"id1"      2541      1781.8     1804.8
"id2"      11.3      621.5      50.2
"id3"      61.2      149.1      22
"id4"      55.3      328.8      97.2
"id5"        183.8       378.3      423.2
!series_matrix_table_end
```

Download the sample file here:
http://www.resources.qiagenbioinformatics.com/madata/GEOSeriesFile.txt

J.2 Affymetrix GeneChip

For Affymetrix, three types of files are currently supported: Affymetrix .CHP files, Affymetrix NetAffx annotation files and tab-delimited pivot or metrics files. Affymetrix .CEL files are currently not supported. However, the Bioconductor R package 'affy' allows you to preprocess the .CEL files and export a txt file containing a table of estimated probe-level expression values in three lines of code:

```
library(affy) # loading Bioconductor library 'affy'
data=ReadAffy() # probe-level data import
eset=rma(data) # probe-level data pre-processing using 'rma'
write.exprs(eset,file="evals.txt") # writing probe expression levels to 'evals-txt'
```

The exported txt file (evals.txt) can be imported into the workbench using the Generic expression data table format importer (see section J.5; you can just 'drag-and-drop' it in). In R, you should have all the CEL files you wish to process in your working directory and the file 'evals.txt' will be written to that directory.

If multiple probes are present for the same gene, further processing may be required to merge them into a single gene-level expression.

J.2.1 Affymetrix CHP expression files

The Affymetrix scanner software produces a number of files when a GeneChip is scanned. Two of these are the .CHP and the .CEL files. These are binary files with native Affymetrix formats. The Affymetrix GeneChips contain a number of probes for each gene (typically between 22 and 40). The .CEL file contains the probe-level intensities, and the .CHP file contains the gene-level information. The gene-level information has been obtained by the scanner software through postprocessing and summarization of the probe-level intensities.

In order to interpret the probe-level information in the .CEL file, the .CDF file for the type of GeneChip that was used is required. Similarly for the .CHP file: in order to interpret the gene-level information in the .CHP file, the .PSI file for the type of GeneChip that was used is required.

In order to import a .CHP file it is required that the corresponding .PSI file is present in the same folder as the .CHP file you want to import, and furthermore, this must be the only .PSI file that is present there. There are no requirements for the name of the .PSI file. Note that the .PSI file itself will not be imported - it is only used to guide the import of the .CHP file which contains the expression values.

Download example .CHP and .PSI files here (note that these are binary files):
http://www.resources.qiagenbioinformatics.com/madata/AffymetrixCHPandPSI.zip

J.2.2 Affymetrix metrix files

The Affymetrix metrics or pivot files are tab-delimited files that may be exported from the Affymetrix scanner software. The metrics files have a lot of technical information that is only partly used in the Workbench. The feature ids (Probe Set Name), expression values (Used Signal), absent/present call (Detection) and absent/present p-value (Detection p-value) are imported into the Workbench.

Download a small example sample file here:
http://www.resources.qiagenbioinformatics.com/madata/AffymetrixMetrics.txt

J.2.3 Affymetrix NetAffx annotation files

The NetAffx annotation files for Whole-Transcript Expression Gene arrays and 3' IVT Expression Analysis Arrays can be imported and used to annotate experiments as shown in section 30.2.3.

Download a small example annotation file here which includes header information:
http://www.resources.qiagenbioinformatics.com/madata/AffymetrixNetAffxAnnotation.csv

J.3 Illumina BeadChip

Both BeadChip expression data files from Illumina's BeadStudio software and the corresponding BeadChip annotation files are supported by *Biomedical Genomics Workbench*. The formats of the BeadStudio and annotation files have changed somewhat over time and various formats are supported.

J.3.1 Illumina expression data, compact format

An example of this format is shown below:

```
TargetID            AVG_Signal          BEAD_STDEV          Detection
GI_10047089-S       112.5               4.2                 0.16903226
GI_10047091-S       127.6               4.8                 0.76774194
```

All this information is imported into the Workbench. The `AVG_Signal` is used as the expression measure.

Download a small sample file here:
http://www.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipCompact.txt

J.3.2 Illumina expression data, extended format

An example of this format is shown below:

```
TargetID        MIN_Signal  AVG_Signal  MAX_Signal  NARRAYS  ARRAY_STDEV  BEAD_STDEV  Avg_NBEADS  Detection
GI_10047089-S   73.7        73.7        73.7        1        NaN          3.4         53          0.05669084
GI_10047091-S   312.7       312.7       312.7       1        NaN          11.1        50          0.99604483
```

All this information is imported into the Workbench. The `AVG_Signal` is used as the expression measure.

Download a small sample file here:
http://www.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipExtended.txt

J.3.3 Illumina expression data, with annotations

An example of this format is shown below:

```
TargetID Accession Symbol Definition Synonym Signal-BG02 DCp32  Detection-BG02 DCp32
GI_10047089-S NM_014332.1 SMPX "Homo sapiens small muscle protein, X-linked (SMPX), mRNA."  -17.6  0.03559657
GI_10047091-S NM_013259.1 NP25 "Homo sapiens neuronal protein (NP25), mRNA." NP22  32.6  0.99604483
GI_10047093-S NM_016299.1 HSP70-4 "Homo sapiens likely ortholog of mouse heat shock protein, 70 kDa 4 (HSP70-4), mRNA."  228.1 1
```

Only the TargetID, Signal and Detection columns will be imported, the remaining columns will be ignored. This means that the annotations are not imported. The `Signal` is used as the expression measure.

Download a small example sample file here:
http://www.resources.qiagenbioinformatics.com/madata/IlluminaBeadStudioWithAnnot
txt

J.3.4 Illumina expression data, multiple samples in one file

This file format has too much information to show it inline in the text. You can download a small example sample file here:
http://www.resources.qiagenbioinformatics.com/madata/IlluminaBeadStudioMultipleS
txt

This file contains data for 18 samples. Each sample has an expression value (the value in the `AVG_Signal` column), a detection p-value, a bead standard deviation and an average bead number column. The workbench recognizes the 18 samples and their columns.

J.3.5 Illumina annotation files

The Workbench supports import of two types of Illumina BeadChip annotation files. These are either comma-separated or tab-delimited .txt files. They can be used to annotate experiments as shown in section 30.2.3.

This file format has too much information to show it inline in the text.

Download a small example annotation file of the first type here:
http://www.resources.qiagenbioinformatics.com/madata/IlluminaBeadChipAnnotation.
txt

J.4 Gene ontology annotation files

The Gene ontology web site provides annotation files for a variety of species which can all be downloaded and imported into the *Biomedical Genomics Workbench*. They can be used to annotate experiments as shown in section 30.2.3. They can also be used with the Gene Set Test and Create Expression Browser tools.

Import GO annotation file using the Standard Import tool. For GO annotation files in GAF format, use the option "Force import as type: Gene Ontology Annotation file" from the drop down menu at the bottom of the Standard Import dialog.

See the complete list of available files, including download links, at http://www.geneontology.
org/GO.current.annotations.shtml.

J.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 *Biomedical Genomics Workbench* as a 'generic' expression or annotation data file. There are a few simple requirements that need to be fulfilled to do this as described below.

J.5.1 Generic expression data table format

The *Biomedical Genomics Workbench* will import a tab, semicolon or comma-separated .txt or .csv file as expression array samples if the following requirements are met:

1. the first non-empty line of the file contains text. All entries, except the first, will be used as sample names
2. the following (non-empty) lines contain the same number of entries as the first non-empty line. The requirements to these are that the first entry should be a string (this will be used as the feature ID) and the remaining entries should contain numbers (which will be used as expression values — one per sample). Empty entries are not allowed, but NaN values are allowed.
3. the file contains at least two samples.

An example of this format is shown below:

```
FeatureID;sample1;sample2;sample3
gene1;200;300;23
gene2;210;30;238
gene3;230;50;23
gene4;50;100;235
gene5;200;300;23
gene6;210;30;238
gene7;230;50;23
gene8;50;100;235
```

This will be imported as three samples with eight genes in each sample.

Download a this example as a file here:
http://www.resources.qiagenbioinformatics.com/madata/CustomExpressionData.txt

J.5.2 Generic annotation file for expression data format

The *Biomedical Genomics Workbench* will import a tab, semicolon or comma-separated .txt or .csv file as an annotation file if the following is met:

1. It has a line which can serve as a valid header line. In order to do this, the line should have a number of headers where at least two are among the valid column headers in the **Column header** column below.

2. It contains one of the PROBE_ID headers (that is: 'Probe Set ID', 'Feature ID', 'ProbeID' or 'Probe_Id').

The importer will import an annotation table with a column for each of the valid column headers (those in the **Column header** column below). Columns with invalid headers will be ignored.

Note that some column headers are alternatives so that only one of the alternative columns headers should be used.

When adding annotations to an experiment, you can specify the column in your annotation file containing the relevant identifiers. These identifiers are matched to the feature ids already present in your experiment. When a match is found, the annotation is added to that entry in the experiment. In other words, at least one column in your annotation file must contain identfiers matching the feature identifiers in the experiment, for those annotations to be applied.

A simple example of an annotation file is shown here:

```
"Probe Set ID","Gene Symbol","Gene Ontology Biological Process"
"1367452_at","Sumo2","0006464 // protein modification process //  not recorded"
"1367453_at","Cdc37","0051726 // regulation of cell cycle //  not recorded"
"1367454_at","Copb2","0006810 // transport //  ///  0016044 // membrane organization // "
```

Download this example plus a more elaborate one here:
http://www.resources.qiagenbioinformatics.com/madata/SimpleCustomAnnotation.csv
http://www.resources.qiagenbioinformatics.com/madata/FullCustomAnnotation.csv

To meet requirements imposed by special functionalities in the workbench, there are a number of further restrictions on the contents in the entries of the columns:

Download sequence functionality In the experiment table, you can click a button to download sequence. This uses the contents of the PUBLIC_ID column, so this column must be present for the action to work and should contain the NCBI accession number.

Annotation tests The annotation tests can make use of several entries in a column as long as a certain format is used. The tests assume that entries are separated by /// and it interprets all that appears before // as the actual entry and all that appears after // within an entry as comments. Example:

```
/// 0000001 //  comment1  /// 0000008 // comment2 /// 0003746 //  comment3
```

The annotation tests will interpret this as three entries (0000001, 0000008, and 0003746) with the according comments.

The most common column headers are summarized below:

Column header in imported file (alternatives separated by commas)	Label in experiment table	Description (tool tip)
Probe Set ID, Feature ID, ProbeID, Probe_Id, transcript_cluster_id	Feature ID	Probe identifier tag
Representative Public ID, Public identifier tag, GenbankAccession	Public identifier tag	Representative public ID
Gene Symbol, GeneSymbol	Gene symbol	Gene symbol
Gene Ontology Biological Process, Ontology_Process, GO_biological_process	GO biological process	Gene Ontology biological process
Gene Ontology Cellular Component, Ontology_Component, GO_cellular_component	GO cellular component	Gene Ontology cellular component
Gene Ontology Molecular Function, Ontology_Function, GO_molecular_function	GO molecular function	Gene Ontology molecular function
Pathway	Pathway	Pathway

The full list of possible column headers:

Column header in imported file (alternatives separated by commas)	Label in experiment table	Description (tool tip)
Species Scientific Name, Species Name, Species	Species name	Scientific species name
GeneChip Array	Gene chip array	Gene Chip Array name
Annotation Date	Annotation date	Date of annotation
Sequence Type	Sequence type	Type of sequence
Sequence Source	Sequence source	Source from which sequence was obtained
Transcript ID(Array Design), Transcript	Transcript ID	Transcript identifier tag
Target Description	Target description	Target description
Archival UniGene Cluster	Archival UniGene cluster	Archival UniGene cluster
UniGene ID, UniGeneID, Unigene_ID, unigene	UniGene ID	UniGene identifier tag
Genome Version	Genome version	Version of genome on which annotation is based
Alignments	Alignments	Alignments
Gene Title	Gene title	Gene title
geng_assignments	Gene assignments	Gene assignments
Chromosomal Location	Chromosomal location	Chromosomal location
Unigene Cluster Type	UniGene cluster type	UniGene cluster type
Ensemble Ensembl	Ensembl	
Entrez Gene, EntrezGeneID, Entrez_Gene_ID	Entrez gene	Entrez gene
SwissProt	SwissProt	SwissProt
EC	EC	EC
OMIM	OMIM	Online Mendelian Inheritance in Man
RefSeq Protein ID	RefSeq protein ID	RefSeq protein identifier tag
RefSeq Transcript ID	RefSeq transcript ID	RefSeq transcript identifier tag
FlyBase	FlyBase	FlyBase
AGI	AGI	AGI
WormBase	WormBase	WormBase
MGI Name	MGI name	MGI name
RGD Name	RGD name	RGD name
SGD accession number	SGD accession number	SGD accession number
InterPro	InterPro	InterPro
Trans Membrane	Trans membrane	Trans membrane
QTL	QTL	QTL
Annotation Description	Annotation description	Annotation description
Annotation Transcript Cluster	Annotation transcript cluster	Annotation transcript cluster
Transcript Assignments	Transcript assignments	Trancript assignments
mrna_assignments	mRNA assignments	mRNA assignments
Annotation Notes	Annotation notes	Annotation notes
GO, Ontology	Go annotations	Go annotations
Cytoband	Cytoband	Cytoband
PrimaryAccession	Primary accession	Primary accession
RefSeqAccession	RefSeq accession	RefSeq accession
GeneName	Gene name	Gene name
TIGRID	TIGR Id	TIGR Id
Description	Description	Description
GenomicCoordinates	Genomic coordinates	Genomic coordinates
Search_key	Search key	Search key
Target	Target	Target
Gid, GI	Genbank identifier	Genbank identifier
Accession	GenBank accession	GenBank accession
Symbol	Gene symbol	Gene symbol
Probe_Type	Probe type	Probe type
crosshyb_type	Crosshyb type	Crosshyb type
category	category	category
Start, Probe_Start	Start	Start
Stop	Stop	Stop
Definition	Definition	Definition
Synonym, Synonyms	Synonym	Synonym
Source	Source	Source
Source_Reference_ID	Source reference id	Source reference id
RefSeq_ID	Reference sequence id	Reference sequence id
ILMN_Gene	Illumina Gene	Illumina Gene
Protein_Product	Protein product	Protein product
protein_domains	Protein domains	Protein domains
Array_Address_Id	Array adress id	Array adress id
Probe_Sequence	Sequence	Sequence
seqname	Seqname	Seqname
Chromosome	Chromosome	Chromosome
strand	Strand	Strand
Probe_Chr_Orientation	Probe chr orientation	Probe chr orientation
Probe_Coordinates	Probe coordinates	Probe coordinates
Obsolete_Probe_Id	Obsolete probe id	Obsolete probe id

Bibliography

[Allison et al., 2006] Allison, D., Cui, X., Page, G., and Sabripour, M. (2006). Microarray data analysis: from disarray to consolidation and consensus. *NATURE REVIEWS GENETICS*, 7(1):55.

[Auer and Doerge, 2010] Auer, P. L. and Doerge, R. (2010). Statistical design and analysis of rna sequencing data. *Genetics*, 185(2):405–416.

[Baggerly et al., 2003] Baggerly, K., Deng, L., Morris, J., and Aldaz, C. (2003). Differential expression in SAGE: accounting for normal between-library variation. *Bioinformatics*, 19(12):1477–1483.

[Benjamini and Hochberg, 1995] Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *JOURNAL-ROYAL STATISTICAL SOCIETY SERIES B*, 57:289–289.

[Berman et al., 2003] Berman, H., Henrick, K., and Nakamura, H. (2003). Announcing the worldwide protein data bank. *Nat Struct Biol*, 10(12):980.

[Bolstad et al., 2003] Bolstad, B., Irizarry, R., Astrand, M., and Speed, T. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193.

[Bommarito et al., 2000] Bommarito, S., Peyret, N., and SantaLucia, J. (2000). Thermodynamic parameters for DNA sequences with dangling ends. *Nucleic Acids Res*, 28(9):1929–1934.

[Choi et al., 2009] Choi, M., Scholl, U. I., Ji, W., Liu, T., Tikhonova, I. R., Zumbo, P., Nayir, A., Bakkaloğlu, A., Özen, S., Sanjad, S., Nelson-Williams, C., Farhi, A., Mane, S., and Lifton, R. P. (2009). Genetic diagnosis by whole exome capture and massively parallel DNA sequencing. *Proc Natl Acad Sci U S A*, 106(45):19096–19101.

[Consortium et al., 2012] Consortium, . G. P., Abecasis, G. R., Auton, A., Brooks, L. D., DePristo, M. A., Durbin, R. M., Handsaker, R. E., Kang, H. M., Marth, G. T., and McVean, G. A. (2012). An integrated map of genetic variation from 1,092 human genomes. *Nature*, 491(7422):56–65.

[Cornette et al., 1987] Cornette, J. L., Cease, K. B., Margalit, H., Spouge, J. L., Berzofsky, J. A., and DeLisi, C. (1987). Hydrophobicity scales and computational techniques for detecting amphipathic structures in proteins. *J Mol Biol*, 195(3):659–685.

[Creighton et al., 2009] Creighton, C. J., Reid, J. G., and Gunaratne, P. H. (2009). Expression profiling of micrornas by deep sequencing. *Brief Bioinform*, 10(5):490–497.

[Cronn et al., 2008] Cronn, R., Liston, A., Parks, M., Gernandt, D. S., Shen, R., and Mockler, T. (2008). Multiplex sequencing of plant chloroplast genomes using solexa sequencing-by-synthesis technology. *Nucleic Acids Res*, 36(19):e122.

[Dudoit et al., 2003] Dudoit, S., Shaffer, J., and Boldrick, J. (2003). Multiple Hypothesis Testing in Microarray Experiments. *STATISTICAL SCIENCE*, 18(1):71–103.

[Eisen et al., 1998] Eisen, M., Spellman, P., Brown, P., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868.

[Eisenberg et al., 1984] Eisenberg, D., Schwarz, E., Komaromy, M., and Wall, R. (1984). Analysis of membrane and surface protein sequences with the hydrophobic moment plot. *J Mol Biol*, 179(1):125–142.

[Emini et al., 1985] Emini, E. A., Hughes, J. V., Perlow, D. S., and Boger, J. (1985). Induction of hepatitis a virus-neutralizing antibody by a virus-specific synthetic peptide. *J Virol*, 55(3):836–839.

[Engelman et al., 1986] Engelman, D. M., Steitz, T. A., and Goldman, A. (1986). Identifying nonpolar transbilayer helices in amino acid sequences of membrane proteins. *Annu Rev Biophys Biophys Chem*, 15:321–353.

[Falcon and Gentleman, 2007] Falcon, S. and Gentleman, R. (2007). Using GOstats to test gene lists for GO term association. *Bioinformatics*, 23(2):257.

[Guo et al., 2006] Guo, L., Lobenhofer, E. K., Wang, C., Shippy, R., Harris, S. C., Zhang, L., Mei, N., Chen, T., Herman, D., Goodsaid, F. M., Hurban, P., Phillips, K. L., Xu, J., Deng, X., Sun, Y. A., Tong, W., Dragan, Y. P., and Shi, L. (2006). Rat toxicogenomic study reveals analytical consistency across microarray platforms. *Nat Biotechnol*, 24(9):1162–1169.

[Heap et al., 2010] Heap, G. A., Yang, J. H. M., Downes, K., Healy, B. C., Hunt, K. A., Bockett, N., Franke, L., Dubois, P. C., Mein, C. A., Dobson, R. J., Albert, T. J., Rodesch, M. J., Clayton, D. G., Todd, J. A., van Heel, D. A., and Plagnol, V. (2010). Genome-wide analysis of allelic expression imbalance in human primary cells by high-throughput transcriptome resequencing. *Hum Mol Genet*, 19(1):122–134.

[Heydarian et al., 2014] Heydarian, M., Romeo Luperchio, T., Cutler, J., Mitchell, C., Kim, M.-S., Pandey, A., Soliner-Webb, B., and Reddy, K. (2014). Prediction of gene activity in early B cell development based on an integrative multi-omics analysis. *J Proteomics Bioinform*, 7(2):050–063.

[Homer N, 2010] Homer N, N. S. (2010). Improved variant discovery through local re-alignment of short-read next-generation sequencing data using srma. *Genome Biol.*, 11(10):R99.

[Hopp and Woods, 1983] Hopp, T. P. and Woods, K. R. (1983). A computer program for predicting protein antigenic determinants. *Mol Immunol*, 20(4):483–489.

[Janin, 1979] Janin, J. (1979). Surface and inside volumes in globular proteins. *Nature*, 277(5696):491–492.

[Kal et al., 1999] Kal, A. J., van Zonneveld, A. J., Benes, V., van den Berg, M., Koerkamp, M. G., Albermann, K., Strack, N., Ruijter, J. M., Richter, A., Dujon, B., Ansorge, W., and Tabak, H. F. (1999). Dynamics of gene expression revealed by comparison of serial analysis of gene expression transcript profiles from yeast grown on two different carbon sources. *Mol Biol Cell*, 10(6):1859–1872.

[Karplus and Schulz, 1985] Karplus, P. A. and Schulz, G. E. (1985). Prediction of chain flexibility in proteins. *Naturwissenschaften*, 72:212–213.

[Kaufman and Rousseeuw, 1990] Kaufman, L. and Rousseeuw, P. (1990). Finding groups in data. an introduction to cluster analysis. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics, New York: Wiley, 1990*.

[Knudsen and Miyamoto, 2003] Knudsen, B. and Miyamoto, M. M. (2003). Sequence alignments and pair hidden markov models using evolutionary history. *Journal of Molecular Biology*, 333(2):453 – 460.

[Kolaskar and Tongaonkar, 1990] Kolaskar, A. S. and Tongaonkar, P. C. (1990). A semi-empirical method for prediction of antigenic determinants on protein antigens. *FEBS Lett*, 276(1-2):172--174.

[Kumar et al., 2013] Kumar, V., Muratani, M., Rayan, N. A., Kraus, P., Lufkin, T., Ng, H. H., and Prabhakar, S. (2013). Uniform, optimal signal processing of mapped deep-sequencing data. *Nat Biotechnol*, 31(7):615–22.

[Kyte and Doolittle, 1982] Kyte, J. and Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *J Mol Biol*, 157(1):105–132.

[Landt et al., 2012] Landt, S. G., Marinov, G. K., Kundaje, A., Kheradpour, P., Pauli, F., Batzoglou, S., Bernstein, B. E., Bickel, P., Brown, J. B., Cayting, P., Chen, Y., DeSalvo, G., Epstein, C., Fisher-Aylor, K. I., Euskirchen, G., Gerstein, M., Gertz, J., Hartemink, A. J., Hoffman, M. M., Iyer, V. R., Jung, Y. L., Karmakar, S., Kellis, M., Kharchenko, P. V., Li, Q., Liu, T., Liu, X. S., Ma, L., Milosavljevic, A., Myers, R. M., Park, P. J., Pazin, M. J., Perry, M. D., Raha, D., Reddy, T. E., Rozowsky, J., Shoresh, N., Sidow, A., Slattery, M., Stamatoyannopoulos, J. A., Tolstorukov, M. Y., White, K. P., Xi, S., Farnham, P. J., Lieb, J. D., Wold, B. J., and Snyder, M. (2012). ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. *Genome Res*, 22(9):1813–31.

[Law et al., 2014] Law, V., Knox, C., Djoumbou, Y., Jewison, T., Guo, A., Liu, Y., Maciejewski, A., Arndt, D., Wilson, M., Neveu, V., Tang, A., Gabriel, G., Ly, C., Adamjee, S., Dame, Z., Han, B., Zhou, Y., and Wishart, D. (2014). Drugbank 4.0: shedding new light on drug metabolism. *Nucleic Acids Res.*, 42:D1091–7.

[Li et al., 2012] Li, J., Lupat, R., Amarasinghe, K. C., Thompson, E. R., Doyle, M. A., Ryland, G. L., Tothill, R. W., Halgamuge, S. K., Campbell, I. G., and Gorringe, K. L. (2012). Contra: copy number analysis for targeted resequencing. *Bioinformatics*, 28(10):1307–1313.

[Lloyd, 1982] Lloyd, S. (1982). Least squares quantization in PCM. *Information Theory, IEEE Transactions on*, 28(2):129–137.

[Lu et al., 2008] Lu, M., Dousis, A. D., and Ma, J. (2008). Opus-rota: A fast and accurate method for side-chain modeling. *Protein Science*, 17(9):1576–1585.

[Marinov et al., 2014] Marinov, G. K., Kundaje, A., Park, P. J., and Wold, B. J. (2014). Large-scale quality analysis of published ChIP-seq data. *G3 (Bethesda)*, 4(2):209–23.

[Martin and Wang, 2011] Martin, J. A. and Wang, Z. (2011). Next-generation transcriptome assembly. *Nat Rev Genet*, 12(10):671–682.

[Meyer et al., 2007] Meyer, M., Stenzel, U., Myles, S., Pruefer, K., and Hofreiter, M. (2007). Targeted high-throughput sequencing of tagged nucleic acid samples. *Nucleic Acids Res*, 35(15):e97.

[Miao et al., 2011] Miao, Z., Cao, Y., and Jiang, T. (2011). Rasp: rapid modeling of protein side chain conformations. *Bioinformatics*, 27(22):3117–3122.

[Morin et al., 2008] Morin, R. D., O'Connor, M. D., Griffith, M., Kuchenbauer, F., Delaney, A., Prabhu, A.-L., Zhao, Y., McDonald, H., Zeng, T., Hirst, M., Eaves, C. J., and Marra, M. A. (2008). Application of massively parallel sequencing to microrna profiling and discovery in human embryonic stem cells. *Genome Res*, 18(4):610–621.

[Morrison, 1968] Morrison, D. R. (1968). Patricia – practical algorithm to retrieve information coded in alphanumeric. *J. ACM*, 15(4):514–534.

[Mortazavi et al., 2008] Mortazavi, A., Williams, B. A., McCue, K., Schaeffer, L., and Wold, B. (2008). Mapping and quantifying mammalian transcriptomes by rna-seq. *Nat Methods*, 5(7):621–628.

[Mukherjee and Zhang, 2009] Mukherjee, S. and Zhang, Y. (2009). MM-align: A quick algorithm for aligning multiple-chain protein complex structures using iterative dynamic programming. *Nucleic Acids Res.*, 37.

[Ng et al., 2009] Ng, S. B., Turner, E. H., Robertson, P. D., Flygare, S. D., Bigham, A. W., Lee, C., Shaffer, T., Wong, M., Bhattacharjee, A., Eichler, E. E., Bamshad, M., Nickerson, D. A., and Shendure, J. (2009). Targeted capture and massively parallel sequencing of 12 human exomes. *Nature*, 461(7261):272–276.

[Nguyen et al., 2011] Nguyen, P., Ma, J., Pei, D., Obert, C., Cheng, C., and Geiger, T. (2011). Identification of errors introduced during high throughput sequencing of the t cell receptor repertoire. *BMC genomics*, 12(1):106.

[Niu and Zhang, 2012] Niu, Y. S. and Zhang, H. (2012). The screening and ranking algorithm to detect dna copy number variations. *Ann Appl Stat*, 6(3):1306–1326.

[Parkhomchuk et al., 2009] Parkhomchuk, D., Borodina, T., Amstislavskiy, V., Banaru, M., Hallen, L., Krobitsch, S., Lehrach, H., and Soldatov, A. (2009). Transcriptome analysis by strand-specific sequencing of complementary dna. *Nucleic Acids Res*, 37(18):e123.

[Robinson et al., 2010] Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140.

[Robinson and Oshlack, 2010] Robinson, M. D. and Oshlack, A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol.*, 11(3):R25.

[Robinson and Smyth, 2007] Robinson, M. D. and Smyth, G. K. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887.

[Robinson and Smyth, 2008] Robinson, M. D. and Smyth, G. K. (2008). Small-sample estimation of negative binomial dispersion, with applications to sage data. *Biostatistics*, 9(2):321–332.

[Rose et al., 1985] Rose, G. D., Geselowitz, A. R., Lesser, G. J., Lee, R. H., and Zehfus, M. H. (1985). Hydrophobicity of amino acid residues in globular proteins. *Science*, 229(4716):834–838.

[Rye et al., 2011] Rye, M. B., Saetrom, P., and Drablos, F. (2011). A manually curated ChIP-seq benchmark demonstrates room for improvement in current peak-finder programs. *Nucleic Acids Res*, 39(4):e25.

[SantaLucia, 1998] SantaLucia, J. (1998). A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *Proc Natl Acad Sci U S A*, 95(4):1460–1465.

[Smith and Waterman, 1981] Smith, T. F. and Waterman, M. S. (1981). Identification of common molecular subsequences. *J Mol Biol*, 147(1):195–197.

[Stanton et al., 2013] Stanton, K. P., Parisi, F., Strino, F., Rabin, N., Asp, P., and Kluger, Y. (2013). Arpeggio: harmonic compression of ChIP-seq data reveals protein-chromatin interaction signatures. *Nucleic Acids Res*, 41(16):e161.

[Stark et al., 2010] Stark, M. S., Tyagi, S., Nancarrow, D. J., Boyle, G. M., Cook, A. L., Whiteman, D. C., Parsons, P. G., Schmidt, C., Sturm, R. A., and Hayward, N. K. (2010). Characterization of the melanoma mirnaome by deep sequencing. *PLoS One*, 5(3):e9685.

[Sturges, 1926] Sturges, H. A. (1926). The choice of a class interval. *Journal of the American Statistical Association*, 21:65–66.

[Tian et al., 2005] Tian, L., Greenberg, S., Kong, S., Altschuler, J., Kohane, I., and Park, P. (2005). Discovering statistically significant pathways in expression profiling studies. *Proceedings of the National Academy of Sciences*, 102(38):13544–13549.

[Tusher et al., 2001] Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci U S A*, 98(9):5116–5121.

[von Ahsen et al., 2001] von Ahsen, N., Wittwer, C. T., and Schütz, E. (2001). Oligonucleotide melting temperatures under PCR conditions: nearest-neighbor corrections for Mg(2+), deoxynucleotide triphosphate, and dimethyl sulfoxide concentrations with comparison to alternative empirical formulas. *Clin Chem*, 47(11):1956–1961.

[Wang et al., 2009] Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet*, 10(1):57–63.

[Welling et al., 1985] Welling, G. W., Weijer, W. J., van der Zee, R., and Welling-Wester, S. (1985). Prediction of sequential antigenic regions in proteins. *FEBS Lett*, 188(2):215–218.

[Wishart et al., 2006] Wishart, D., Knox, C., Guo, A., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z., and Woolsey, J. (2006). Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Res.*, 34:D668–72.

[Wyman et al., 2009] Wyman, S. K., Parkin, R. K., Mitchell, P. S., Fritz, B. R., O'Briant, K., Godwin, A. K., Urban, N., Drescher, C. W., Knudsen, B. S., and Tewari, M. (2009). Repertoire of micrornas in epithelial ovarian cancer as determined by next generation sequencing of small rna cdna libraries. *PLoS One*, 4(4):e5311.

[Xu and Zhang, 2010] Xu, J. and Zhang, Y. (2010). How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics*, 26(7):889–95.

[Zhang and Skolnick, 2004] Zhang, Y. and Skolnick, J. (2004). Scoring function for automated assessment of protein structure template quality. *Proteins*, 57(4):702–10.

Part XIII

Index

Appendix K

Index

Index