CLC **Microbial Genomics** Module

USER MANUAL

User manual for

CLC Microbial Genomics Module 24.0.1

Windows, macOS and Linux

March 11, 2024

This software is for research purposes only.

QIAGEN Aarhus
Silkeborgvej 2
Prismet
DK-8000 Aarhus C
Denmark

Contents

Part I

Introduction

Chapter 1

Introduction

Welcome to *CLC Microbial Genomics Module 24.0.1* – a software package supporting your daily bioinformatics work.

1.1 The concept of CLC Microbial Genomics Module

CLC Microbial Genomics Module includes tools for microbial community analysis as well as tools for epidemiological typing of microbial isolates.

Microbiome composition analysis based on 16S rRNA and other commonly used metagenome derived amplicon data is fully supported. The primary output of the clustering, tallying and taxonomic assignment processes is an OTU abundance table that lists the abundances of OTUs in the samples under investigation. In addition, analyses based on whole shotgun metagenomic data are also available, leading to taxonomic profiling abundance tables. CLC Microbial Genomics Module also offers the possibility to investigate biological functions associated with complex communities using Gene Ontology (GO) and Pfam databases to annotate whole shotgun metagenomic data in functional abundance tables. All abundance tables are viewable through a number of intuitive visualization options. Secondary analyses include estimations of alpha and beta diversities, in addition to various statistical tests for differential abundance.

Tools for NGS-MLST typing and identification of antimicrobial resistance genes are included in CLC Microbial Genomics Module to enable epidemiological typing of microbial isolates using NGS data. In cases when the precise identity of the isolated species is not known, the tool automatically detects the most closely related reference genome in NCBI's RefSeq bacterial genome collection and the corresponding MLST scheme from MLST.net or PubMLST.org. The powerful new CLC metadata framework allows fast and intuitive browsing, sorting, filtering and selection of samples and associated metadata, including results obtained during analysis. This metadata framework provides a dashboard-like overview for easy filtering and selection of samples for other analyses such as k-mer or SNP tree reconstruction and visualisation for outbreak analysis.

For convenience, expert-configured workflows for microbiome analysis as well as epidemiological typing allow the user to get from raw NGS reads through data processing and statistical analysis to the final graphical results in very few steps. Reference databases and MLST schemes needed to perform the analyses are automatically downloadable using dedicated tools, and can be easily customized to fit the specific needs of your research.

Unless specified otherwise, all tools described in this manual can be found in the Microbial Genomics Module folder that will be placed in the Toolbox once the plugin is installed.

The CLC Microbial Genomics Module is frequently updated. A detailed list of new features, improvements, bug fixes, and changes is available at https://digitalinsights.qiagen.com/clc-microbial-genomics-module-latest-improvements/.

1.2 Contact information

CLC Microbial Genomics Module is developed by:

QIAGEN Aarhus
Silkeborgvej 2
Prismet
8000 Aarhus C
Denmark

https://digitalinsights.qiagen.com/

Email: ts-bioinformatics@qiagen.com

The QIAGEN Aarhus team continuously improves products with your interests in mind. We welcome feedback and suggestions for new features or improvements. How to contact us is described at: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Contact_information_citation.html

You can also make use of our online documentation resources, including:

- Core product manuals https://digitalinsights.qiagen.com/technical-support/manuals/
- Plugin manuals https://digitalinsights.qiagen.com/products-overview/plugins/
- Tutorials https://digitalinsights.qiagen.com/support/tutorials/
- Frequently Asked Questions https://qiagen.my.salesforce-sites.com/KnowledgeBase/KnowledgeNavigatorPage

1.3 System requirements

CLC Microbial Genomics Module 24.0.1 is for use with CLC Genomics Workbench version 24.0.1 or newer. The system requirements for CLC Genomics Workbench are provided at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=System_requirements.html.

The system requirements for CLC Microbial Genomics Module are the same as those for CLC Genomics Workbench, except for the following:

- An AMD/Intel CPU that supports AVX2 or an Apple M series CPU is required for the tools below:

- Annotate with DIAMOND
- Annotate CDS with Best DIAMOND Hits
- Find Resistance with ShortBRED
- Type with MLST Scheme

Special requirements for the MLST Scheme tools

The system requirement for the MLST Scheme tools depends on the size of the MLST schemes (both the number of alleles and the number of sequence types). A laptop with 16GB of memory is normally sufficient for 7-gene schemes or cgMLST schemes based on a moderate number of isolates. Downloading and constructing or typing with larger schemes may require more memory, and in general we recommend at least 64GB of memory when working with cg/wgMLST schemes based on more than 100 isolates.

Special requirements for OTU Clustering

The memory requirement of *Reference based* OTU clustering depends on the size of the reference database used; more and longer sequences require more run time and memory. Newer version of common choices (e.g. the full SILVA SSU database) are likely to be too large for a 16 GB machine. Instead we recommend using clustered databases (e.g. the SILVA SSU 99% database) and/or otherwise filtering and subsetting the database, to minimize its size.

Special requirements for Classify Long Read Amplicons

The memory requirement for **Classify Long Read Amplicons** depends on both sample size and the size of reference database. The following are examples of maximum memory usage given different sample and database sizes.

Sample size	Database size	Memory usage
100,000 reads	9 MB / 50,000 sequences	15 GB
100,000 reads	13 MB / 27,000 sequences	10 GB
1,000,000 reads	9 MB / 50,000 sequences	25 GB
1,000,000 reads	13 MB / 27,000 sequences MB	20 GB

Large reference databases like the unclustered SILVA SSU database, are expected to require more than 32GB of available memory.

Special requirements for Taxonomic Profiling

The performance of the Taxonomic Profiling tool depends on the reference database used - the more complete a database, the better the taxonomic profiling. However, running Taxonomic Profiling with a given database size will require at least the same amount of memory. For example, a 14 GB database requires at least 16 GB of RAM, and a 56 GB database requires a minimum of 64 GB RAM. When creating your reference database with the Download Custom Microbial Reference Database tool, you will get a warning about the memory requirements needed for running the Taxonomic Profiling tool with this database.

Special requirements for De Novo Assemble Metagenome

At least 16 GB RAM is recommended for running De Novo Assemble Metagenome.

1.4 Installing modules

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

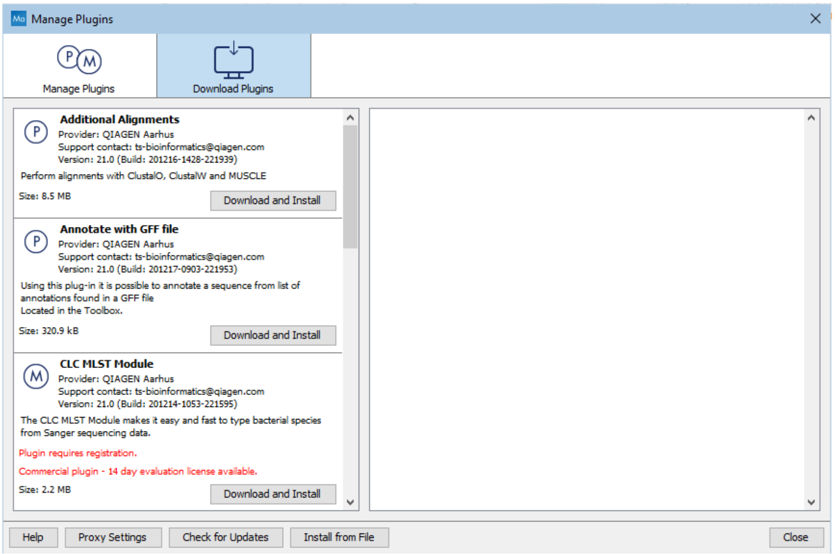
Plugins and modules are installed and uninstalled using the Workbench Plugin Manager. To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

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

The Plugin Manager has two tabs at the top:

- **Manage Plugins** An overview of your installed plugins and modules is provided under this tab.
- **Download Plugins** Plugins and modules available to download and install are listed in this tab.

To install a plugin, click on the **Download Plugins** tab (figure 1.1). Select a plugin. Information about it will be shown in the right hand panel. Click on the **Download and Install** button to install the plugin.



Additional Alignments
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 21.0 (Build: 201216-1428-221939)
Perform alignments with ClustalO, ClustalW and MUSCLE
Size: 8.5 MB

Annotate with GFF file
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 21.0 (Build: 201217-0903-221953)
Using this plug-in it is possible to annotate a sequence from list of annotations found in a GFF file
Located in the Toolbox.
Size: 320.9 kB

CLC MLST Module
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 21.0 (Build: 201214-1053-221595)
The CLC MLST Module makes it easy and fast to type bacterial species from Sanger sequencing data.
Plugin requires registration.
Commercial plugin - 14 day evaluation license available.
Size: 2.2 MB

Figure 1.1: *Plugins and modules available for installation are listed in the Plugin Manager under the Download Plugins tab.*

Accepting the license agreement

The End User License Agreement (EULA) must be read and accepted as part of the installation process. Please read the EULA text carefully, and if you agree to it, check the box next to the text **I accept these terms**. If further information is requested from you, please fill this in before clicking on the **Finish** button.

Installing a cpa file

If you have a .cpa installer file for CLC Microbial Genomics Module, you can install it by clicking on the **Install from File** button at the bottom of the Plugin Manager.

If you are working on a system not connected to the internet, plugin and module .cpa files can be downloaded from https://digitalinsights.qiagen.com/products-overview/plugins/using a networked machine, and then transferred to the non-networked machine for installation.

Restart to complete the installation

Newly installed plugins and modules will be available for use after restarting the software. When you close the Plugin Manager, a dialog appears offering the opportunity to restart the *CLC Workbench*.

1.4.1 Licensing modules

When you have installed the CLC Microbial Genomics Module and start a tool from that module for the first time, the License Assistant will open (figure 1.2).

The License Assistant can also be launched by opening the Workbench Plugin Manager, selecting the installed module from under the Manage Plugins tab, and clicking on the button labeled *Import License*.

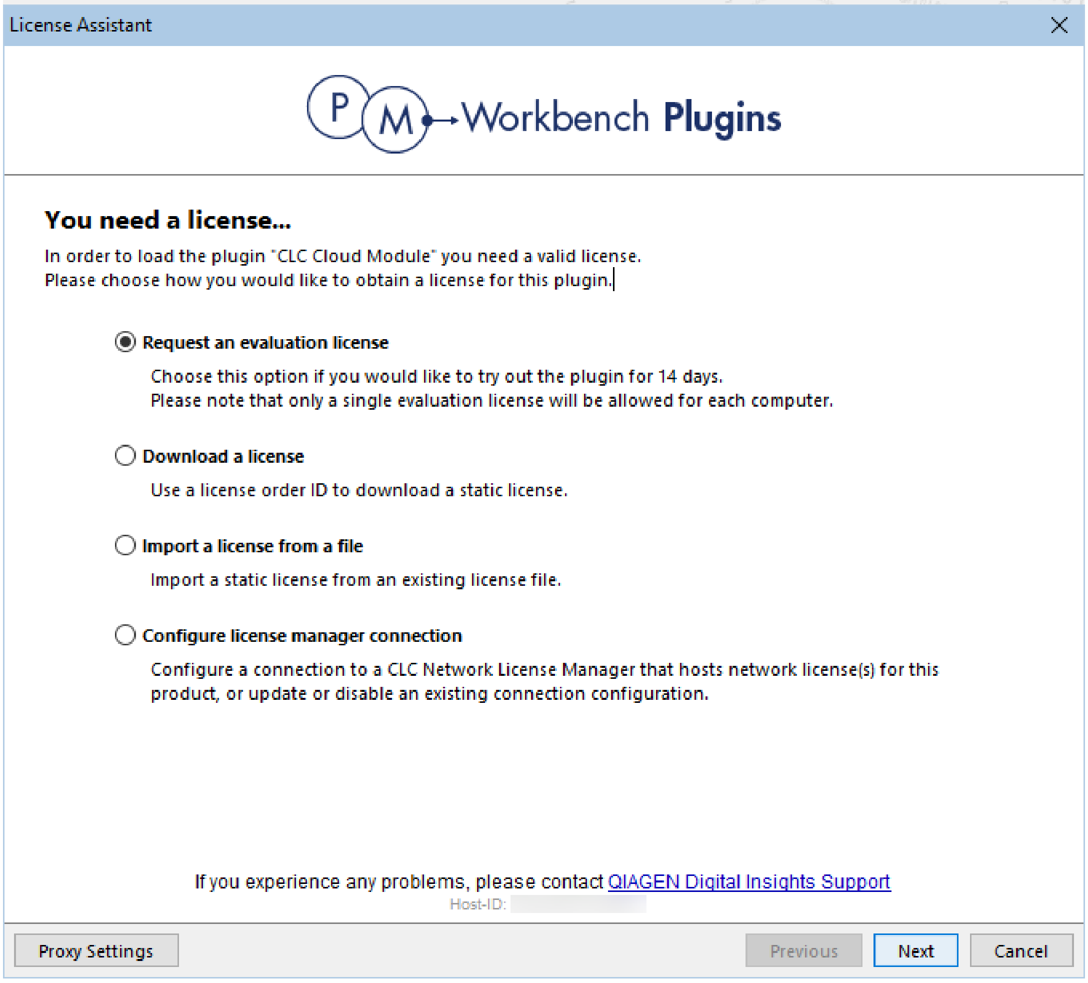
To install a license, the *CLC Workbench* must be run in administrator mode. On Windows, you can do this by right-clicking the program shortcut and choosing "Run as Administrator". On Linux and Mac, it means you must launch the program such that it is run by an administrative user.

The following options are available:

- **Request an evaluation license**. Request a fully functional, time-limited license.
- **Download a license**. Use the license order ID received when you purchased 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.
- **Configure license manager connection**. If your organization has a *CLC Network License Manager*, select this option to configure the connection to it.

These options are described in detail in sections under https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Workbench_Licenses.html.

To download licenses, including evaluation licenses, your machine must have access to the external network. To install licenses on non-networked machines, please see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Download_static_license_on_non_networked_machine.html.



You need a license...

In order to load the plugin "CLC Cloud Module" you need a valid license.
Please choose how you would like to obtain a license for this plugin.

● **Request an evaluation license**

Choose this option if you would like to try out the plugin for 14 days.
Please note that only a single evaluation license will be allowed for each computer.

○ **Download a license**

Use a license order ID to download a static license.

○ **Import a license from a file**

Import a static license from an existing license file.

○ **Configure license manager connection**

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

If you experience any problems, please contact QIAGEN Digital Insights Support

Host-ID:

Figure 1.2: *The License Assistant provides options for licensing modules installed on the Workbench.*

1.4.2 Uninstalling modules

Plugins and modules are uninstalled using the Workbench Plugin Manager.To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

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

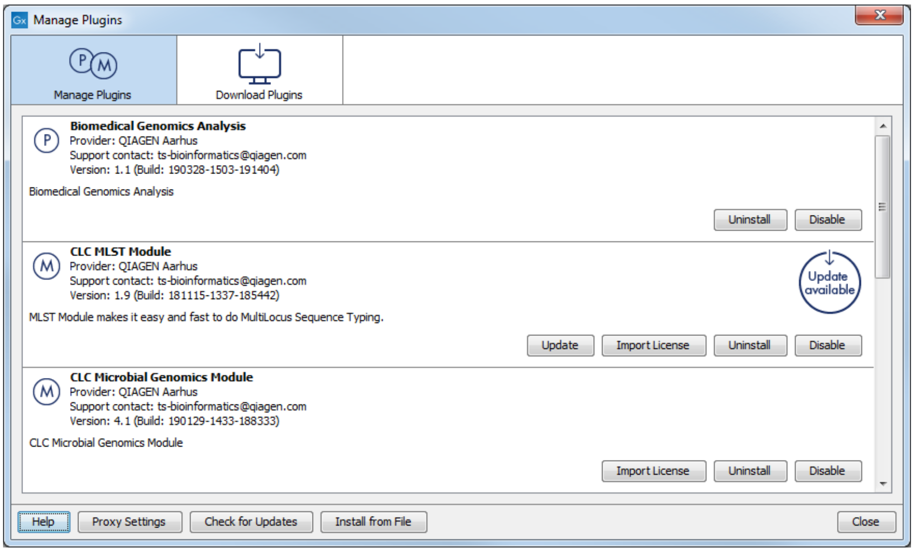
This will open the Plugin Manager (figure 1.3). Installed plugins and modules are shown under the Manage Plugins tab of the Plugins Manager.

To uninstall a plugin or module, click on its entry in the list, and click on the **Uninstall** button.

Plugins and modules are not uninstalled until the Workbench is restarted. When you close the Plugin Manager, a dialog appears offering the opportunity to restart the *CLC Workbench*.

Disabling a plugin without uninstalling it

If you do not want a plugin to be loaded the next time you start the Workbench, select it in the list under the Manage Plugins tab and click on the **Disable** button.



Biomedical Genomics Analysis
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 1.1 (Build: 190328-1503-191404)

Biomedical Genomics Analysis

CLC MLST Module
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 1.9 (Build: 181115-1337-185442)

MLST Module makes it easy and fast to do MultiLocus Sequence Typing.

CLC Microbial Genomics Module
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 4.1 (Build: 190129-1433-188333)

CLC Microbial Genomics Module

Figure 1.3: *Installed plugins and modules are listed in the Plugins Manager under the Manage Plugins tab.*

1.5 Installing server extensions

To use the tools and functionalities of CLC Microbial Genomics Module on a *CLC Server*:

1. You need to purchase a license to run tools delivered by the CLC Microbial Genomics Server Extension.
2. A *CLC Server* administrator must install the license on the single server, or on the master node in a job node or grid node setup, as described in section 1.5.1.
3. A *CLC Server* administrator must install the CLC Microbial Genomics Server Extension on the *CLC Server*, as described below.

Download and install server plugins and server extensions

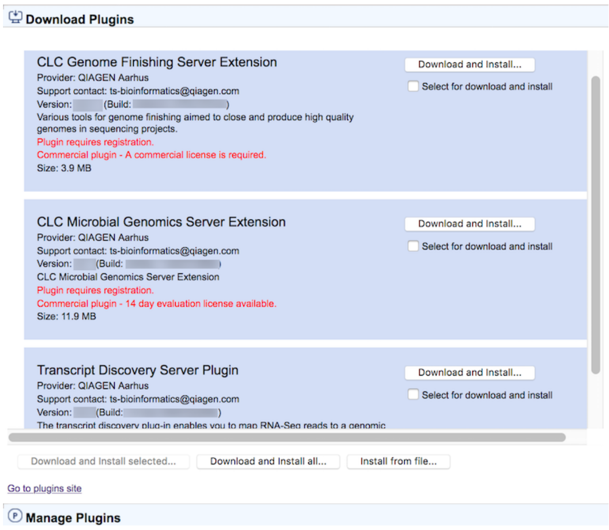
Plugins, including server extensions (commercial plugins), are installed by going to the **Extensions ( ) ** tab in the web administrative interface of the single server, or the master node of a job node or grid nod setup, and opening the **Download Plugins ( )** area (figure 1.4).

If the machine has access to the external network, plugins can be both downloaded and installed via the *CLC Server* administrative interface. To do this, locate the plugin in the list under the **Download Plugins** ( ) area and click on the **Download and Install...** button.

To download and install multiple plugins at once on a networked machine, check the "Select for download and install" box beside each relevant plugin, and then click on the **Download and Install All...** button.

If you are working on a machine without access to the external network, server plugin (.cpa) files can be downloaded from: https://digitalinsights.qiagen.com/products-overview/plugins/ and installed by browsing for the downloaded file and clicking on the **Install from File...** button.

The *CLC Server* must be restarted to complete the installation or removal of plugins and server extensions. All jobs still in the queue at the time the server is shut down will be dropped and



Download Plugins
Download and Install...
☐ Select for download and install
Download and Install...
☐ Select for download and install
Download and Install...
☐ Select for download and install
Download and Install selected...
Download and Install all...
Install from file...
Go to plugins site
Manage Plugins

CLC Genome Finishing Server Extension
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: [redacted] (Build: [redacted])
Various tools for genome finishing aimed to close and produce high quality genomes in sequencing projects.
Plugin requires registration.
Commercial plugin - A commercial license is required.
Size: 3.9 MB

CLC Microbial Genomics Server Extension
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: [redacted] (Build: [redacted])
CLC Microbial Genomics Server Extension
Plugin requires registration.
Commercial plugin - 14 day evaluation license available.
Size: 11.9 MB

Transcript Discovery Server Plugin
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: [redacted] (Build: [redacted])
The transcript discovery plug-in enables you to map RNA-Seq reads to a genomic

Figure 1.4: *Installing plugins and server extensions is done in the Download Plugins area under the Extensions tab.*

would need to be resubmitted. To minimize the impact on users, the server can be put into Maintenance Mode. In brief: running in Maintenance Mode allows current jobs to run, but no new jobs to be submitted, and users cannot log in. The *CLC Server* can then be restarted when desired. Each time you install or remove a plugin, you will be offered the opportunity to enter Maintenance Mode. You will also be offered the option to restart the *CLC Server*. If you choose not to restart when prompted, you can restart later using the option under the **Server maintenance (** ⚙ **)** tab.

For job node setups only:

- Once the *master CLC Server* is up and running normally, then restart each *job node CLC Server* so that the plugin is ready to run on each node. This is handled for you if you restart the server using the functionality under

  Management (👥 **)** | **Server maintenance (** ⚙ **)**

- In the web administrative interface on the *master* CLC Server, check that the plugin is enabled for each job node.

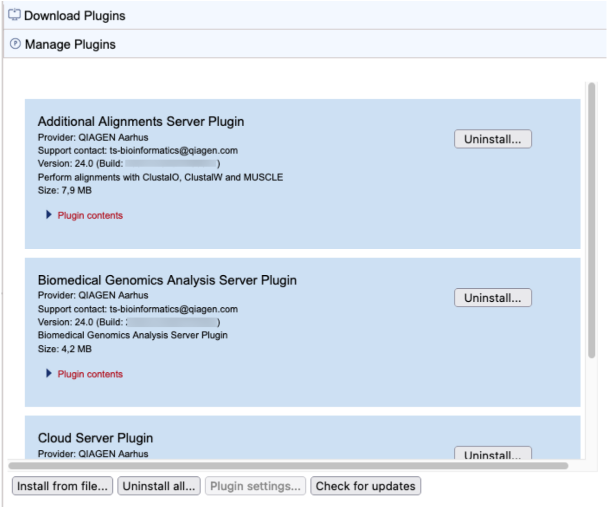
Installation and updating of plugins on connected job nodes requires that direct data transfer from client systems has been enabled, which is done by the *CLC Server* administrator, under the "External data" tab.

Grid workers will be re-deployed when a plugin is installed on the master server. Thus, no further action is needed to enable the newly installed plugin to be used on grid nodes.

Managing installed server plugins

Installed plugins can be updated or uninstalled, from under the **Manage Plugins** (Ⓟ) area (figure 1.5), under the **Extensions (** ⧉ **)** tab.

The list of tools delivered with a server plugin can be seen by clicking on the **Plugin contents** link to expand that section. Workflows delivered with a server plugin are not shown in this listing.



Additional Alignments Server Plugin
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 24.0 (Build: [redacted])
Perform alignments with ClustalO, ClustalW and MUSCLE
Size: 7,9 MB
Plugin contents

Biomedical Genomics Analysis Server Plugin
Provider: QIAGEN Aarhus
Support contact: ts-bioinformatics@qiagen.com
Version: 24.0 (Build: [redacted])
Biomedical Genomics Analysis Server Plugin
Size: 4,2 MB
Plugin contents

Cloud Server Plugin
Provider: QIAGEN Aarhus

Figure 1.5: *Managing installed plugins and server extensions is done in the Manage Plugins area under the Extensions tab. Clicking on Plugin contents opens a list of the tools delivered by the plugin.*

Links to related documentation

- Logging into the *CLC Server* web administrative interface: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Logging_into_administrative_interface.html
- Maintenance Mode: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Server_maintenance.html
- Restarting the server: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Starting_stopping_server.html
- Plugins on job node setups: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Installing_Server_plugins_on_job_nodes.html
- Grid worker re-deployment: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Overview_Model_II.html

Plugin compatibility with the server software

The version of plugins and server extensions installed must be compatible with the version of the *CLC Server* being run. A message is written under an installed plugin's name if it is not compatible with the version of the *CLC Server* software running.

When upgrading to a new major version of the *CLC Server*, all plugins will need to be updated. This means removing the old version and installing a new version.

Incompatibilities can also arise when updating to a new bug fix or minor feature release of the *CLC Server*. We recommend opening the **Manage Plugins** area after any server software upgrade to check for messages about the installed plugins.

Licensing server extensions is described in section 1.5.1.

1.5.1 Licensing server extensions

Licenses are installed on a single server or on the master node of a job node or grid node setup.

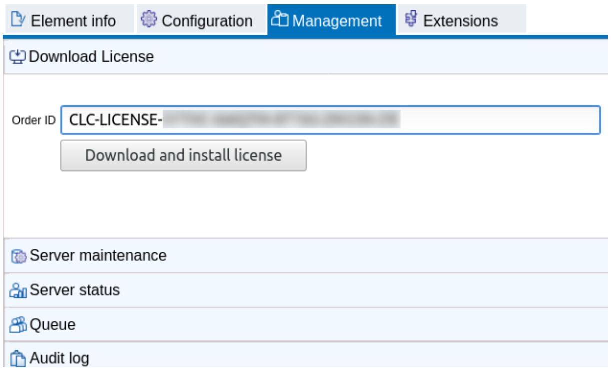
To download and install a license:

- Log into the web administrative interface of the single server or master node as an administrative user.
- Under the **Management ()** tab, open the **Download License ()** tab.
- Enter the Order ID supplied by QIAGEN into the Order ID field and click on the "Download and Install License..." button (figure 1.6).

Please contact ts-bioinformatics@qiagen.com if you have not received an Order ID.

The *CLC Server* must be restarted for new license files to be loaded. Details about restarting can be found at `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Starting_stopping_server.html`.

Each time you download a license file, a new file is created in the `licenses` folder under the *CLC Server* installation area. *If you are upgrading* an existing license file, *delete the old file* from this area before restarting.



Order ID CLC-LICENSE-[redacted]

Figure 1.6: *License management in done under the Management tab tab.*

Part II

Core Functionalities

Chapter 2

Microbial template workflows

Template workflows are provided as example workflows. They can be launched as they are from the Toolbox, or copies can be easily opened, allowing you to optimize the workflow to fit your specific application.

To open a template workflow to view the design or edit it, you can:

- Right-click on the workflow name in the Toolbox in the lower, left side of the Workbench under:

  Toolbox | **Template Workflows**

  and select the option **Open Copy of Workflow** from the right-click menu.

  or

- Open the Workflow Manager by clicking on the **Workflows** button ( ) in the top toolbar, and choose **Manage Workflows**.

  Click on the **Template Workflows** tab and then select the workflow you wish to edit. Then click on the **Open Copy of Workflow** button.

For an introduction to workflows and information on how to configure workflow elements, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Workflows.html.

In the following sections, we describe the template workflows distributed with CLC Microbial Genomics Module.

2.1 Taxonomic Analysis template workflows

The Taxonomic Analysis template workflows are found at:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis ( )**

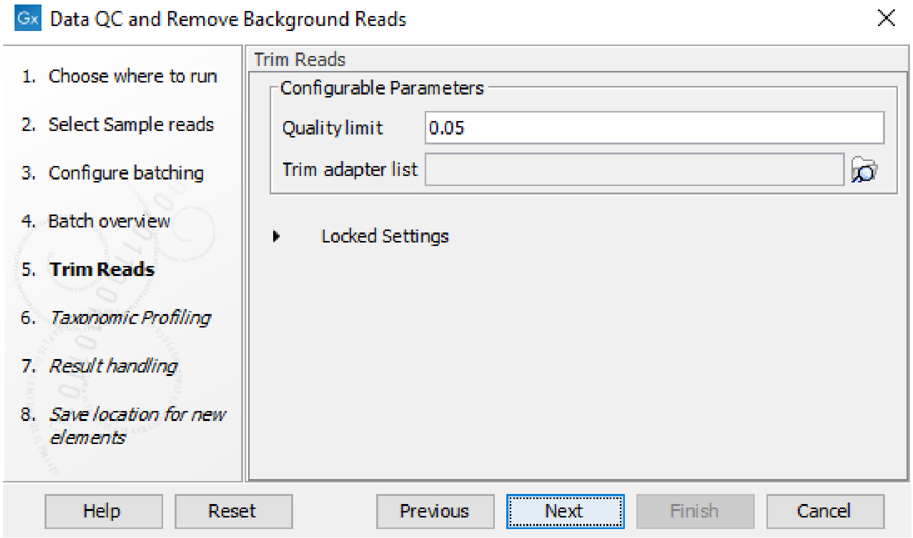
2.1.1 Data QC and Remove Background Reads

The **Data QC and Remove Background Reads** workflow performs trimming of reads, creates a QC report and cleans the dataset from background DNA, leaving back only the reads that match the reference genome(s).

To run the Data QC and Remove Background Reads workflow, go to:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Metagenomics** () | **Taxonomic Analysis** () | **Data QC and Remove Background Reads** ()

In the "Trim Sequences" dialog, you can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters (figure 2.1).



Data QC and Remove Background Reads

1. Choose where to run
2. Select Sample reads
3. Configure batching
4. Batch overview
5. **Trim Reads**
6. *Taxonomic Profiling*
7. *Result handling*
8. *Save location for new elements*

Trim Reads

Configurable Parameters

Quality limit: 0.05

Trim adapter list:

Locked Settings

Figure 2.1: *You can choose to trim adapter sequences from your sequencing reads.*

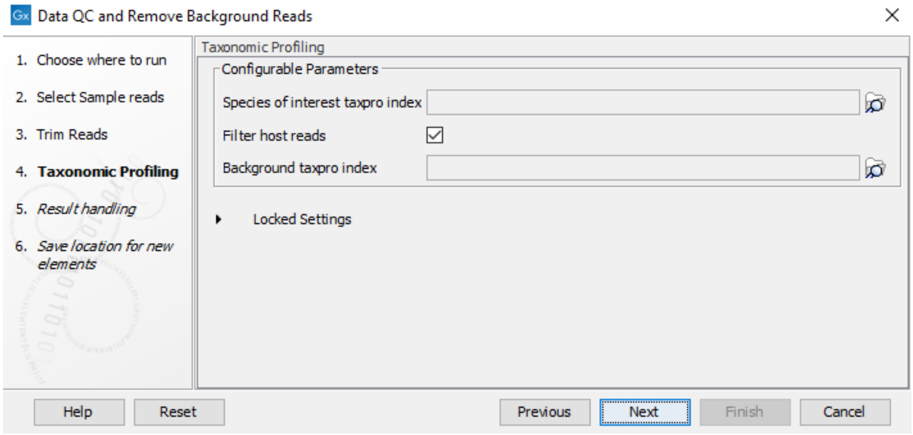
The parameters that can be set are:

- **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.
- **Trim adapter list**: the adapter sequences to trim (if any).

In the Taxonomic Profiling dialog, select the "Species of interest taxpro index" you will use to map the reads (figure 2.2). Here, you can also choose to "Filter background reads". You must then specify the "Background taxpro index" (in the case of human microbiota, the Homo sapiens GRCh38 for example). The reference database can be obtained by using the Download Curated Microbial Reference Database tool (section 15.1) or Download Custom Microbial Reference Database tool (section 15.2). The host index and (if using the custom downloader) the microbial reference index are built with the Create Taxonomic Profiling Index tool (section 15.4).

The workflow will output three folders:

- **Cleaned reads**: trimmed reads mapping to the species of interest taxpro index of reference genome(s).
- **Background reads**: reads mapping to the background taxpro index.



Data QC and Remove Background Reads

1. Choose where to run
2. Select Sample reads
3. Trim Reads
4. **Taxonomic Profiling**
5. *Result handling*
6. *Save location for new elements*

Taxonomic Profiling

Configurable Parameters

Species of interest taxpro index

Filter host reads ☑

Background taxpro index

Locked Settings

Figure 2.2: *Select the reference databaes, and potentially a background taxpro index to remove possible contamination.*

- **Unmapped reads**: reads not mapping to the species of interest or the background taxpro index.

In addition, it generates three reports: a trimming report, a graphical QC report and a supplementary QC report. All of these should be inspected in order to determine whether the quality of the sequencing reads and the trimming are acceptable. For a detailed description of the QC reports and indication on how to interpret the different values, see `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html`.

For the trimming report, see `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html`.

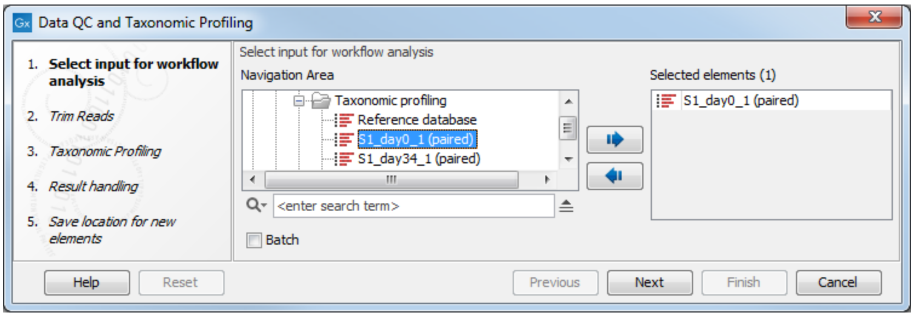
2.1.2 Data QC and Taxonomic Profiling

The Data QC and Taxonomic Profiling combines the Taxonomic Profiling tool with a trimming step and additionally creates sequencing QC reports. The workflow outputs a taxonomic profiling abundance table as well as additional reports on the trimming, QC and taxonomic analysis.

To run the tool, go to:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Data QC and Taxonomic Profiling** ( )

You can select only one read file to analyze (figure 2.3). Alternatively, multiple files can be run using the batch mode.



Data QC and Taxonomic Profiling

1. **Select input for workflow analysis**
2. *Trim Reads*
3. *Taxonomic Profiling*
4. *Result handling*
5. *Save location for new elements*

Select input for workflow analysis

Navigation Area

Taxonomic profiling
Reference database
S1_day0_1 (paired)
S1_day34_1 (paired)

<enter search term>

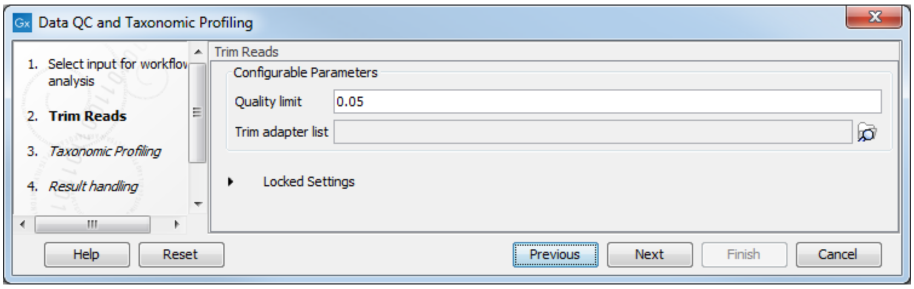
□ Batch

Selected elements (1)

S1_day0_1 (paired)

Figure 2.3: *Select the reads to analyze.*

In the "Trim Sequences" dialog, you can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 2.4).



Data QC and Taxonomic Profiling

1. Select input for workflow analysis
2. **Trim Reads**
3. *Taxonomic Profiling*
4. *Result handling*

Trim Reads

Configurable Parameters

Quality limit: 0.05

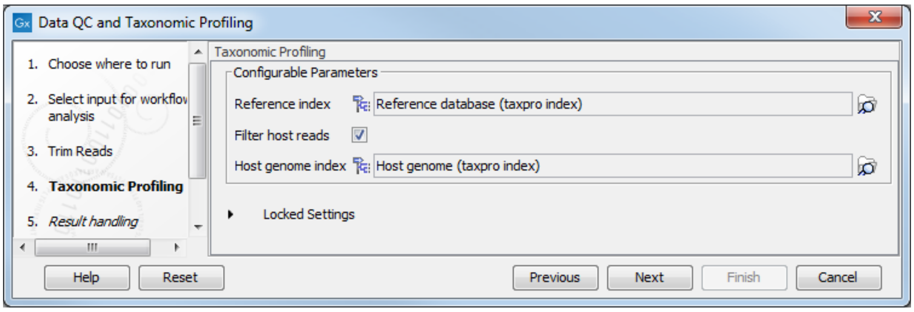
Trim adapter list:

Figure 2.4: *You can choose to trim adapter sequences from your sequencing reads.*

The parameters that can be set are:

- **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.
- **Also search on reversed sequence**: the adapter sequences will also be searched on reverse sequences.

In the "Taxonomic Profiling" dialog (figure 2.5), choose the list of references that you wish to map the reads against. You could also remove host DNA by specifying a reference genome for the host (in the case of human microbiota, the Homo sapiens GRCh38 for example). The reference database can be obtained by using the Download Curated Microbial Reference Database tool (section 15.1) or Download Custom Microbial Reference Database tool (section 15.2). The host index and (if using the custom downloader) the microbial reference index are built with the Create Taxonomic Profiling Index tool (section 15.4).



Data QC and Taxonomic Profiling

1. Choose where to run
2. Select input for workflow analysis
3. Trim Reads
4. **Taxonomic Profiling**
5. *Result handling*

Taxonomic Profiling

Configurable Parameters

Reference index: Reference database (taxpro index)

Filter host reads: ☑

Host genome index: Host genome (taxpro index)

Figure 2.5: *Specify the reference database. You can also check the option "Filter host reads" and specify the host genome.*

The abundance table displays the names of the identified taxa (possibly with their underlying assemblies), along with their full taxonomy, the total amount of reads associated with that taxon, the total number of reads associated with the children of that taxon, and a coverage estimate. The table can be visualized using the Stacked bar charts and stacked area charts function, as well as the Sunburst charts (see section 5.2.3).

The Taxonomic Profiling report is divided in three sections:

- **Taxonomic Summary** Includes information about which taxonomic levels were found in the data sample and how many different taxa were found on each level.

- **Classification of reads** Summarizes the number of reads in the sample and the number of uniquely mapping reads.
- **Reference database Summary** Information on the applied reference database. A subsection is added in case any duplicates are found in the database - these are not used when constructing the taxonomic profile.

In addition, it generates three reports: a trimming report, a graphical QC report and a supplementary QC report. All of these should be inspected in order to determine whether the quality of the sequencing reads and the trimming are acceptable. For a detailed description of the QC reports and indication on how to interpret the different values, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html.

For the trimming report, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html.

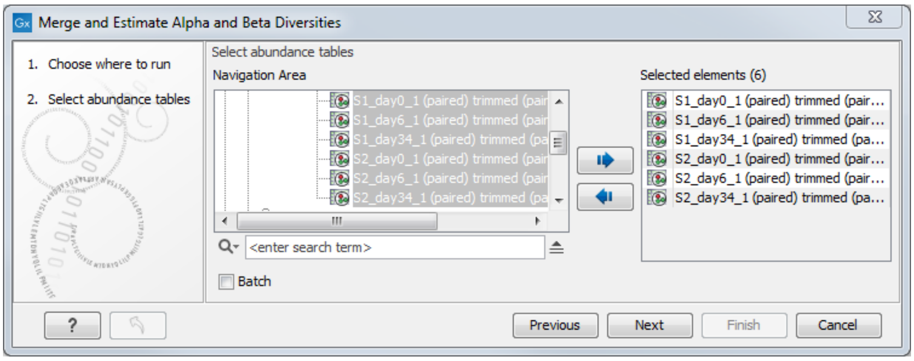
2.1.3 Merge and Estimate Alpha and Beta diversities

The **Merge and Estimate Alpha and Beta diversities** workflow requires several abundance tables as input file. The first tool of the workflow is the **Merge Abundance Tables**. The output is a single merged abundance table that will be used as input for two additional tools, the **Alpha diversity** tool and the **Beta diversity** tool. Running this workflow will therefore give three outputs: a diversity report for the alpha diversity, a PCoA for the beta diversity and a merged abundance table.

To run the tool, go to:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Merge and Estimate Alpha and Beta diversities** ( )

In the first step, select several abundance tables (figure 2.6).

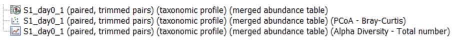


S1_day0_1 (paired) trimmed (pair...
S1_day6_1 (paired) trimmed (pair...
S1_day34_1 (paired) trimmed (pa...
S2_day0_1 (paired) trimmed (pair...
S2_day6_1 (paired) trimmed (pair...
S2_day34_1 (paired) trimmed (pa...

Figure 2.6: *Select abundance tables.*

In the second and third steps, you can choose parameters for the Alpha Diversity and for the Beta Diversity analyses. The parameters are described in section 6.3 and section 6.4.

The Merge and Estimate Alpha and Beta Diversities workflow generates the results seen in figure 2.7.



S1_day0_1 (paired, trimmed pairs) (taxonomic profile) (merged abundance table)
S1_day0_1 (paired, trimmed pairs) (taxonomic profile) (merged abundance table) (PCoA - Bray-Curtis)
S1_day0_1 (paired, trimmed pairs) (taxonomic profile) (merged abundance table) (Alpha Diversity - Total number)

Figure 2.7: *Results from the Merge and Estimate Alpha and Beta Diversities workflow.*

Please refer to section 6.3 and section 6.4 to learn more about interpreting these results.

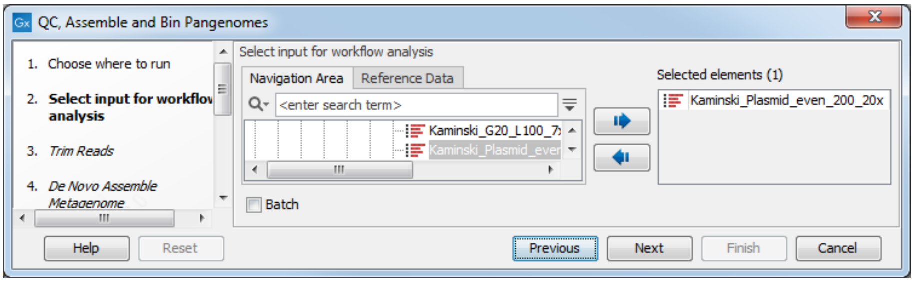
2.1.4 QC, Assemble and Bin Pangenomes

This workflow guides the investigator through the key steps to analyze whole-genome shotgun metagenomic reads and deconstruct them into clusters of sequences (bins) using the tools Bin Pangenomes by Taxonomy and Bin Pangenomes by Sequence. The inputs to the workflow are short reads belonging to a single metagenome sample (also split in multiple sequence objects). The outputs are two sequence lists objects: one for reads and one for assembled contigs, labelled for bin association. Reports are also output at each step.

To run the workflow, go to:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **QC, Assemble and Bin Pangenomes** ( )

In the first step, one or more reads sequence objects are selected (figure 2.8).



QC, Assemble and Bin Pangenomes

1. Choose where to run
2. **Select input for workflow analysis**
3. *Trim Reads*
4. *De Novo Assemble Metagenome*

Select input for workflow analysis

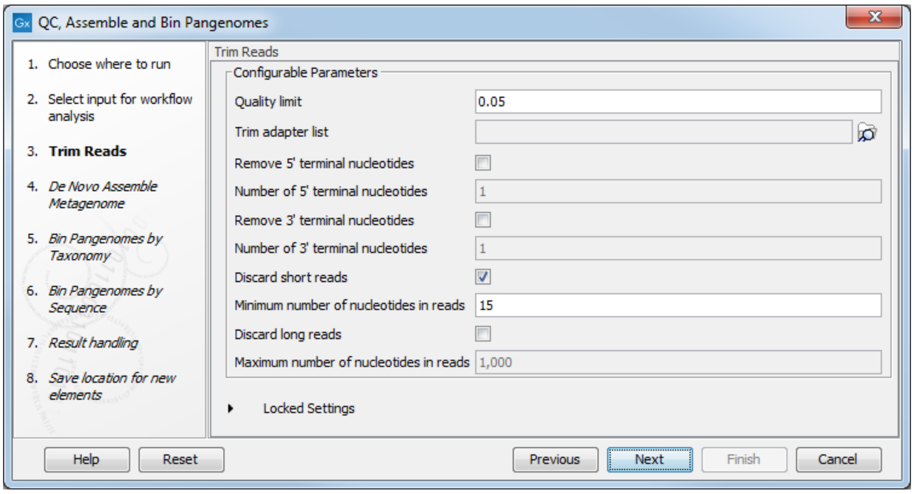
Kaminski_G20_L100_7
Kaminski_Plasmid_even

□ Batch

Selected elements (1)
Kaminski_Plasmid_even_200_20x

Figure 2.8: *Select the reads.*

The workflow first performs QC of raw reads using basic quality-based trimming, but fixed-length trimming can also be added. In the "Trim Reads" dialog, you can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 2.9).



QC, Assemble and Bin Pangenomes

1. Choose where to run
2. Select input for workflow analysis
3. **Trim Reads**
4. *De Novo Assemble Metagenome*
5. *Bin Pangenomes by Taxonomy*
6. *Bin Pangenomes by Sequence*
7. *Result handling*
8. *Save location for new elements*

Trim Reads

Configurable Parameters

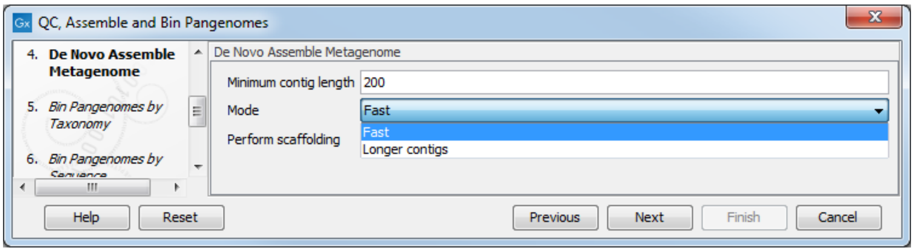
Quality limit	0.05
Trim adapter list	
Remove 5' terminal nucleotides	□
Number of 5' terminal nucleotides	1
Remove 3' terminal nucleotides	□
Number of 3' terminal nucleotides	1
Discard short reads	☑
Minimum number of nucleotides in reads	15
Discard long reads	□
Maximum number of nucleotides in reads	1,000

▸ Locked Settings

Figure 2.9: *Trim the reads.*

In the next step, QC-processed reads are assembled in contigs using the De novo Assemble Metagenome tool. Specify minimum contig length, the type of de novo assembly you wish to perform (fast, or optimized for longer contigs), and whether you wish to perform scaffolding (figure 2.10).

Reads and contigs are then first binned according to taxonomic association and then based on


QC, Assemble and Bin Pangenomes
4. De Novo Assemble Metagenome
5. Bin Pangenomes by Taxonomy
6. Bin Pangenomes by Sequence
De Novo Assemble Metagenome
Help
Reset
Previous
Next
Finish
Cancel

Minimum contig length: 200

Mode: Fast (options: Fast, Longer contigs)

Perform scaffolding

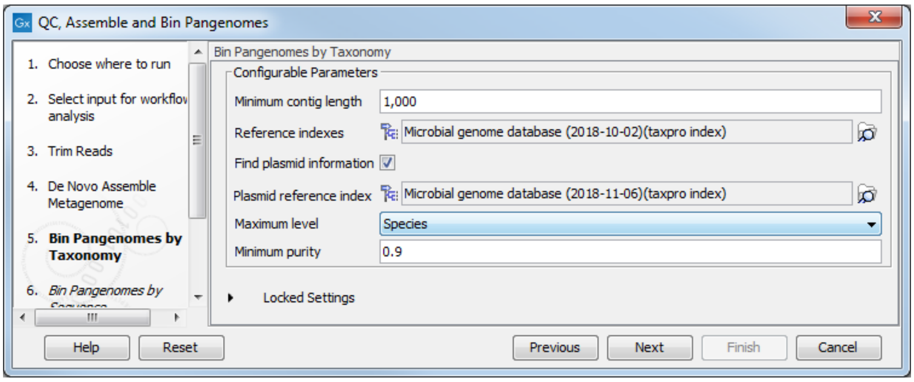
Figure 2.10: *Parameters for the De Novo Assembly Metagenome tool.*

sequence similarity. The tool is designed to work on contigs assembled from the same set of reads used as input (as in the workflow, see section 2.1.4). The tool will use the result of the De novo assembly configured here, but you can set the minimum contig length needed in the next dialog (figure 2.11).

As reference databases, one or two Taxonomic Profiling index files can be provided:

- the file provided as "Reference indexes" is used to find taxonomic information for the reads
- the file provided as "Plasmid reference index" (once the "Find plasmid information option is checked) is used to distinguish genomic reads from plasmid reads.

Both references can be obtained by using the Download Curated Microbial Reference Database tool (section 15.1) or Download Custom Microbial Reference Database tool (section 15.2). If using the custom downloader, the microbial reference index is built with the Create Taxonomic Profiling Index tool (section 15.4).


QC, Assemble and Bin Pangenomes
1. Choose where to run
2. Select input for workflow analysis
3. Trim Reads
4. De Novo Assemble Metagenome
5. Bin Pangenomes by Taxonomy
6. Bin Pangenomes by Sequence
Bin Pangenomes by Taxonomy
Configurable Parameters
Locked Settings
Help
Reset
Previous
Next
Finish
Cancel

Minimum contig length: 1,000

Reference indexes: Microbial genome database (2018-10-02)(taxpro index)

Find plasmid information: ☑

Plasmid reference index: Microbial genome database (2018-11-06)(taxpro index)

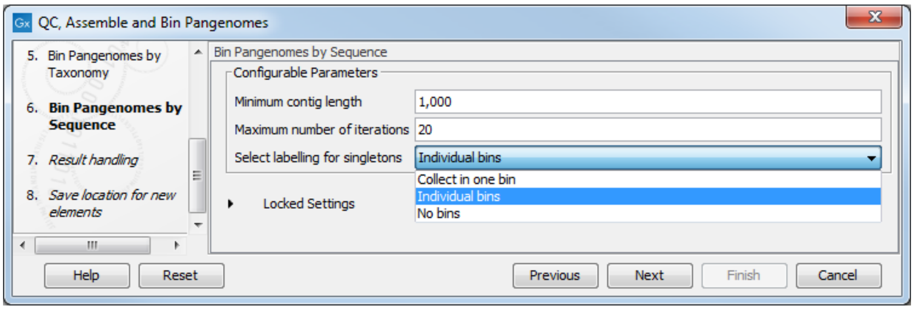
Maximum level: Species

Minimum purity: 0.9

Figure 2.11: *Select the references and configure the Bin Pangenomes by Taxonomy.*

Depending on the dataset, it may be necessary to adapt the contig purity settings, where "Maximum level" refers to a maximum level in the taxonomic tree and where a specific "Minimum purity" per contig needs to be reached in order for it to be considered a part of a bin. For example, if Maximum level = Genus and Minimum purity = 0.8 and 512 reads map to a given contig, at least 0.8 * 512 = 410 reads need to have the same Genus level taxonomy in order for the contig to become part of the respective bin. If more precise taxonomic information is available (e.g., on Species level) with the requested minimum purity, this information will be used instead.

In the next dialog (figure 2.12), configure the parameters for the Bin Pangenomes by Sequence: once again you can set the minimum contig length of the contigs generated by the De Novo Assembly Metagenome tool. You can also choose the maximum of iterations that should be performed, and how to label singletons (bins with only one genome).


QC, Assemble and Bin Pangenomes

5. Bin Pangenomes by Taxonomy
6. **Bin Pangenomes by Sequence**
7. *Result handling*
8. *Save location for new elements*

Bin Pangenomes by Sequence

Configurable Parameters

Minimum contig length: 1,000

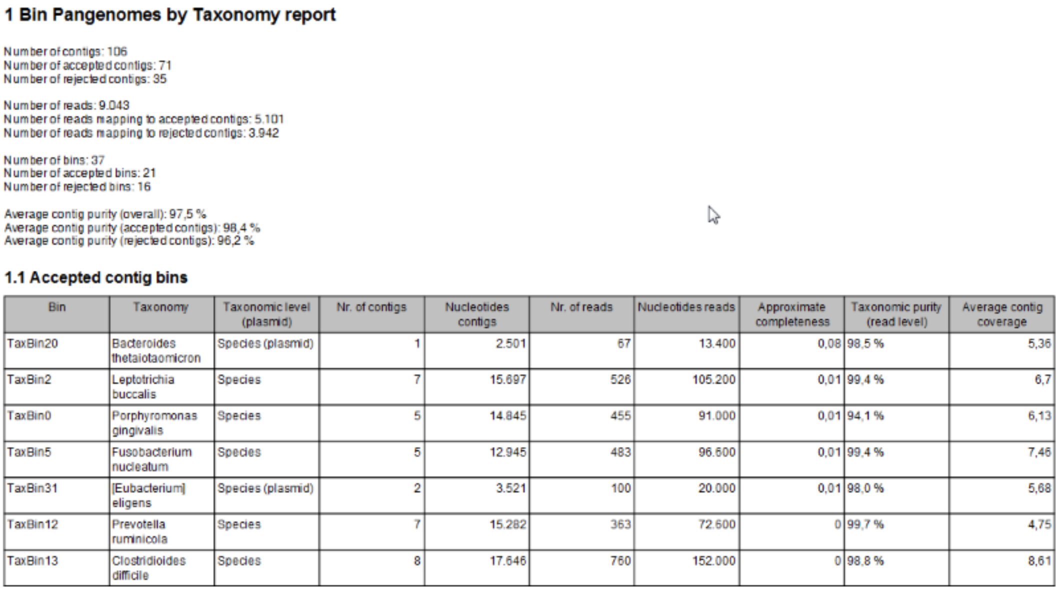
Maximum number of iterations: 20

Select labelling for singletons: Individual bins

Figure 2.12: *Configure the Bin Pangenomes by Sequence.*

The tool will produce the following outputs:

- QC graphic report and QC supplementary report. For a detailed description of the QC reports and indication on how to interpret the different values, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html.
- An assembly summary report (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbenc current/index.php?manual=De_novo_assembly_report.html.
- A Bin Pangenomes by Taxonomy report (figure 2.13). Sequences with the same TaxBin label should have very similar, if not identical, taxonomy.
- A Bin Pangenomes by Sequence (figure 2.14). Sequences with same Bin label are closely related in sequence space.
- A list of reads, and a list of contigs, listing binned (and unbinned) reads and contigs from both binning steps.


1 Bin Pangenomes by Taxonomy report

Number of contigs: 106
Number of accepted contigs: 71
Number of rejected contigs: 35

Number of reads: 9.043
Number of reads mapping to accepted contigs: 5.101
Number of reads mapping to rejected contigs: 3.942

Number of bins: 37
Number of accepted bins: 21
Number of rejected bins: 16

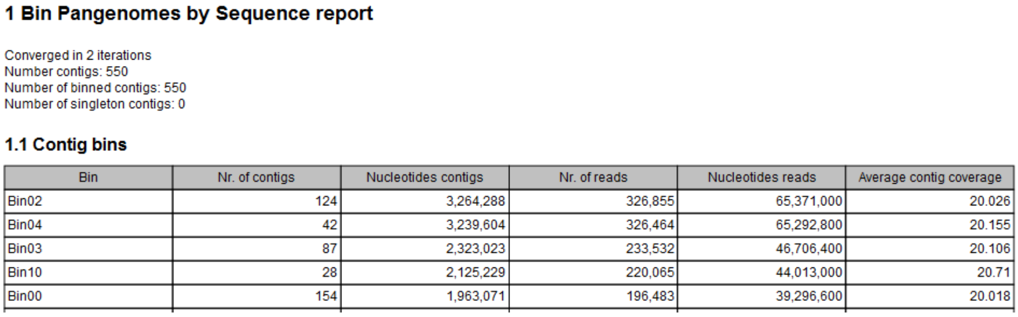
Average contig purity (overall): 97,5 %
Average contig purity (accepted contigs): 98,4 %
Average contig purity (rejected contigs): 96,2 %

1.1 Accepted contig bins

Bin	Taxonomy	Taxonomic level (plasmid)	Nr. of contigs	Nucleotides contigs	Nr. of reads	Nucleotides reads	Approximate completeness	Taxonomic purity (read level)	Average contig coverage
TaxBin20	Bacteroides thetaiotaomicron	Species (plasmid)	1	2.501	67	13.400	0,08	98,5 %	5,36
TaxBin2	Leptotrichia buccalis	Species	7	15.697	526	105.200	0,01	99,4 %	6,7
TaxBin0	Porphyromonas gingivalis	Species	5	14.845	455	91.000	0,01	94,1 %	6,13
TaxBin5	Fusobacterium nucleatum	Species	5	12.945	483	96.600	0,01	99,4 %	7,46
TaxBin31	[Eubacterium] eligens	Species (plasmid)	2	3.521	100	20.000	0,01	98,0 %	5,68
TaxBin12	Prevotella ruminicola	Species	7	15.282	363	72.600	0	99,7 %	4,75
TaxBin13	Clostridioides difficile	Species	8	17.646	760	152.000	0	98,8 %	8,61

Figure 2.13: *The Bin Pangenomes by Taxonomy report. Sequences with the same TaxBin label should have very similar, if not identical, taxonomy.*

Individual bins can be extracted from the sequence and contig lists (when seen as tables, you can find the bin label inthe Assembly_ID column) and used for downstream analysis such as reference-based assembly (or re-assembly), functional analysis, typing etc.



1 Bin Pangenomes by Sequence report

Converged in 2 iterations
Number contigs: 550
Number of binned contigs: 550
Number of singleton contigs: 0

1.1 Contig bins

Bin	Nr. of contigs	Nucleotides contigs	Nr. of reads	Nucleotides reads	Average contig coverage
Bin02	124	3,264,288	326,855	65,371,000	20.026
Bin04	42	3,239,604	326,464	65,292,800	20.155
Bin03	87	2,323,023	233,532	46,706,400	20.106
Bin10	28	2,125,229	220,065	44,013,000	20.71
Bin00	154	1,963,071	196,483	39,296,600	20.018

Figure 2.14: *The Bin Pangenomes by Sequence report. Sequences with same Bin label are closely related in sequence space.*

2.2 Amplicon-Based Analysis template workflows

The Amplicon-Based Analysis template workflows are found at:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Metagenomics** () | **Amplicon-Based Analysis** ()

2.2.1 Data QC and OTU Clustering

The **Data QC and OTU Clustering** workflow consists of 3 tools being executed sequentially (figure 2.15). The only necessary input to run the workflow are the reads you want to cluster. You also have the option to provide a list of the primers that were used to sequence these reads if you wish to perform the adapters trimming step with the **Trim Sequences** tool.

The first tool is the **Trim Reads** tool. Together with the sequencing primer list, this tool provides a list of trimmed sequences that will be the input of the **Filter Samples Based on Number of Reads** tool. The results of the trimming and the filter steps are compiled in two reports. The "filtered" list of reads (devoid of reads of poor quality) will be used for the final tool of the workflow, the **OTU clustering** tool. This tool will give a report, and an abundance table with the newly created OTUs, their abundance at each site as well as the total abundance for all samples.

2.2.2 Detect Amplicon Sequence Variants and Assign Taxonomies

The Detect Amplicon Sequence Variants and Assign Taxonomies workflow processes reads from amplicon sequencing to yield a merged multi-sample (if applicable) ASV abundance table, and subsequently assigns taxonomies to the ASVs (amplicon sequence variants).

We recommend making preliminary evaluations of the read lengths and qualities, to decide on parameter settings like read length. This can be done by running a single sample through the workflow, and taking a look at the resulting trim report section *Read length before / after trimming*.

Launching the workflow

The Detect Amplicon Sequence Variants and Assign Taxonomies workflow is at:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **Detect Amplicon Sequence Variants and Assign Taxonomies** ()

Launch the workflow and step through the wizard.

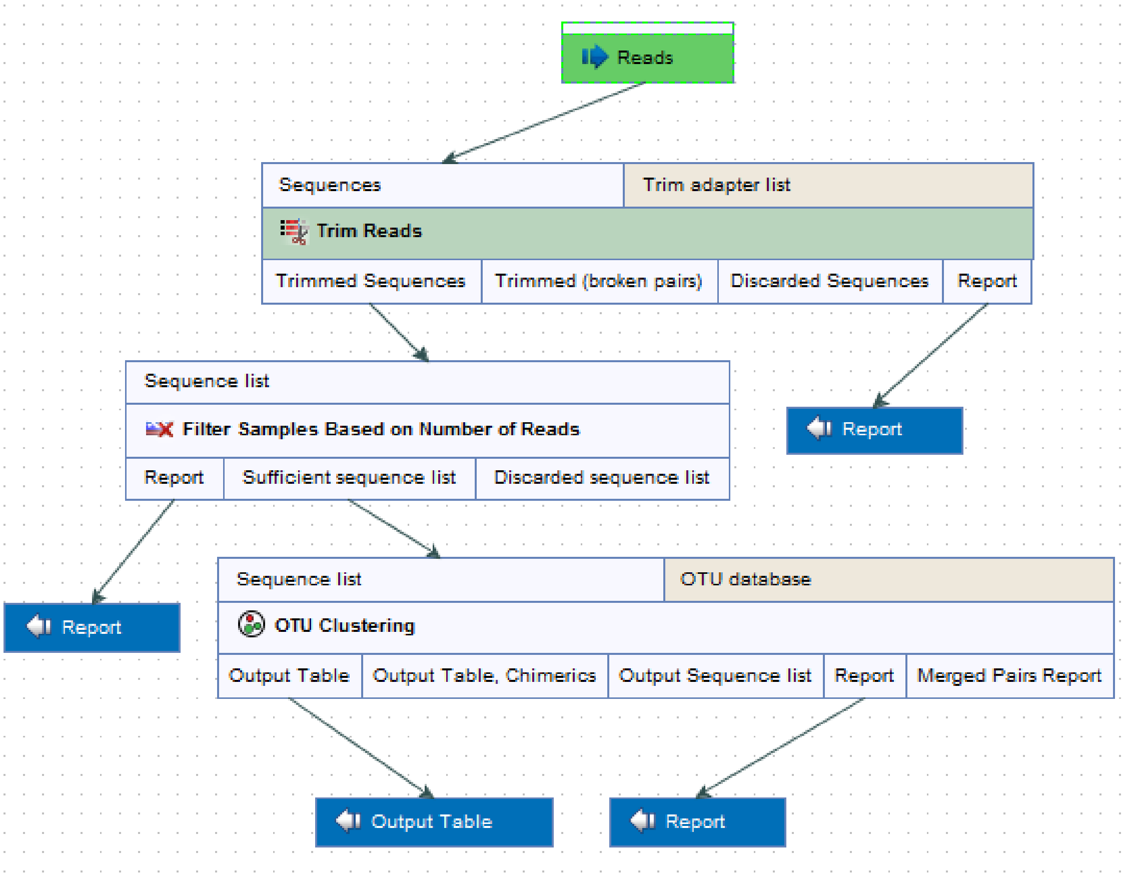
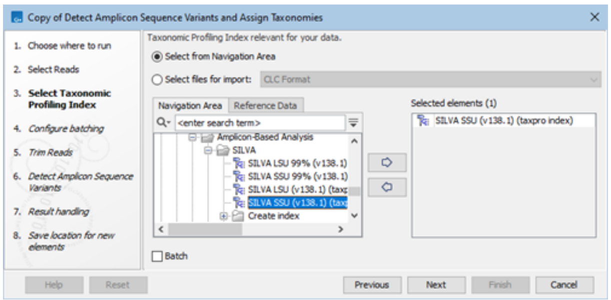


Figure 2.15: *Layout of the Data QC and OTU clustering workflow.*

1. Select the sequence list(s) containing the reads to process and click on **Next**.

2. Select a *Taxonomic Profiling Index* and click on **Next** (figure 2.16).


Copy of Detect Amplicon Sequence Variants and Assign Taxonomies

1. Choose where to run
2. Select Reads
3. **Select Taxonomic Profiling Index**
4. *Configure batching*
5. *Trim Reads*
6. *Detect Amplicon Sequence Variants*
7. *Result handling*
8. *Save location for new elements*

Taxonomic Profiling Index relevant for your data.

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area | Reference Data

<enter search term>

- Amplicon-Based Analysis
  - SILVA
    - SILVA LSU 99% (v138.1)
    - SILVA SSU 99% (v138.1)
    - SILVA LSU (v138.1) (taxp
    - SILVA SSU (v138.1) (taxp
  - Create index

Selected elements (1)

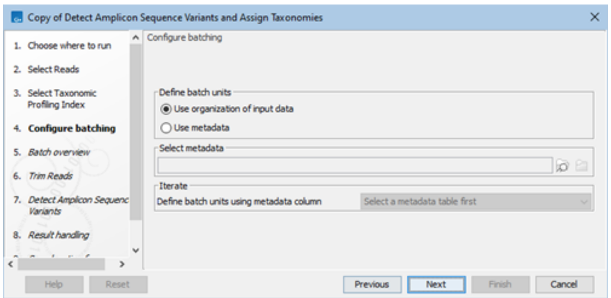
SILVA SSU (v138.1) (taxpro index)

□ Batch

Figure 2.16: *Wizard step for selecting the Taxonomic Profiling Index.*

3. The 'Configure batching' and 'Batch overview' steps can be left as is (figure 2.17), or configured as described in `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Launching_workflows_individually_in_batches.html`.

4. Select a *Trim Adapter List* if relevant for your application (figure 2.18). The *Trim Adapter List* should correspond to the adapters used for sequencing. If no input is provided, the tool will



Configure batching

Define batch units

● Use organization of input data

○ Use metadata

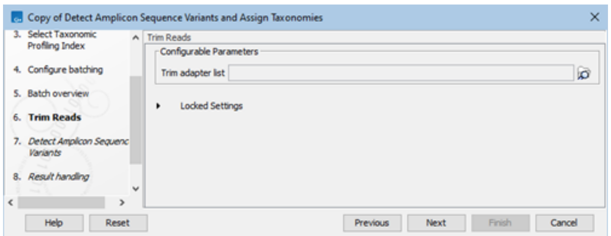
Select metadata

Iterate

Define batch units using metadata column: Select a metadata table first

Figure 2.17: *Optional wizard step to configure metadata for the input sequences.*

skip the adapter trimming step. Click on **Next**.


Copy of Detect Amplicon Sequence Variants and Assign Taxonomies
3. Select Taxonomic Profiling Index
4. Configure batching
5. Batch overview
6. Trim Reads
7. Detect Amplicon Sequenc Variants
8. Result handling
Locked Settings
Help
Reset
Previous
Next
Finish
Cancel

Trim Reads

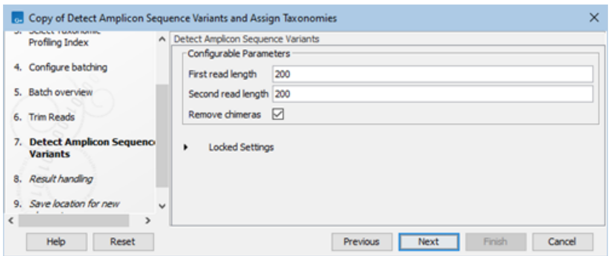
Configurable Parameters

Trim adapter list:

Figure 2.18: *Wizard step for selecting the Trim Adapter List.*

5. Choose the trim length to use for **Detect Amplicon Sequence Variants** and decide whether to remove chimeras by toggling the *Remove chimeras* box (figure 2.19). Click on **Next**.

   The optimal read length setting will depend on the length of your reads after trimming. We recommend that you have a look at the trim report section *Read length before / after trimming* if you are unsure about what value to set.



Detect Amplicon Sequence Variants

Configurable Parameters

First read length: 200

Second read length: 200

Remove chimeras: ☑

Figure 2.19: *Wizard step for selecting read trim length and whether to remove chimeras in the Detect Amplicon Sequence Variants tool.*

6. Finally, select a location to save outputs to and click on **Finish**.

Workflow tools and outputs

The Detect Amplicon Sequence Variants and Assign Taxonomies workflow consists of the below mentioned tools. See figure 2.20 for a full overview of the workflow.

- **QC for Sequencing Reads.** Performs basic QC on the sequencing reads and outputs a report that can be used to evaluate the quality of the sequencing reads, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html. A graphical and a supplementary report is output for each input sample.

- **Trim Reads**. Removes adapter sequences (optional) and low quality nucleotides, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_Reads.html. The tool outputs a trim report for each sample.
- **Detect Amplicon Sequence Variants**. Trims and filters the reads, followed by dereplication and denoising. If chosen in the wizard, chimeras will be removed, and in case of paired-end reads, the unique read pairs are merged, see section 4.6. The workflow outputs one ASV sequence list and ASV report per sample.
- **Merge Abundance Tables**. Merges the per-sample ASV abundance tables from the previous step and outputs a merge report. The merged ASV table is used as input for **Assign Taxonomies to Sequences in Abundance Table**.
- **Assign Taxonomies to Sequences in Abundance Table**. Assigns taxonomies to the sequences of the merged ASV abundance table according to the reference index provided, see section 6.2. The tool outputs a report and a merged ASV abundance table with taxonomies.
  To learn about the ASV abundance table, see section 4.6.2.
- **Combined analysis report**. Combines the report content from the Trim Reads and Detect Amplicon Sequence Variants tools for all samples in the workflow run.
- **Combined QC report**. Contains QC metrics for the raw reads for all samples in the workflow run.

2.2.3 Estimate Alpha and Beta Diversities

The **Estimate Alpha and Beta Diversities** workflow consists of 5 tools and requires only the OTU table as input file (figure 2.21).

Remember to add metadata to the abundance table before starting the workflow. Adding metadata can be done very early on, by importing metadata and associating reads to it before generating an OTU abundance table with the OTU Clustering tool or the Data QC and OTU Clustering workflow. The metadata will propagate to the abundance table automatically. When working with reads that were not associated with metadata in the first place, it is always possible to add metadata to an already existing abundance table with the tool Add Metadata to Abundance Table.

The first tool of the workflow is the **Filter OTUs Based on the Number of Reads**. The output is a reduced abundance table that will be used as input for three other tools:

- **Align OTUs with MUSCLE**, a tool that will produce an alignment used to reconstruct a Maximum Likelihood Phylogeny (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Maximum_Likelihood_Phylogeny.html), which will in turn output a phylogenetic tree also used as input in the following two tools.
- **Alpha diversity** tool
- **Beta diversity** tool

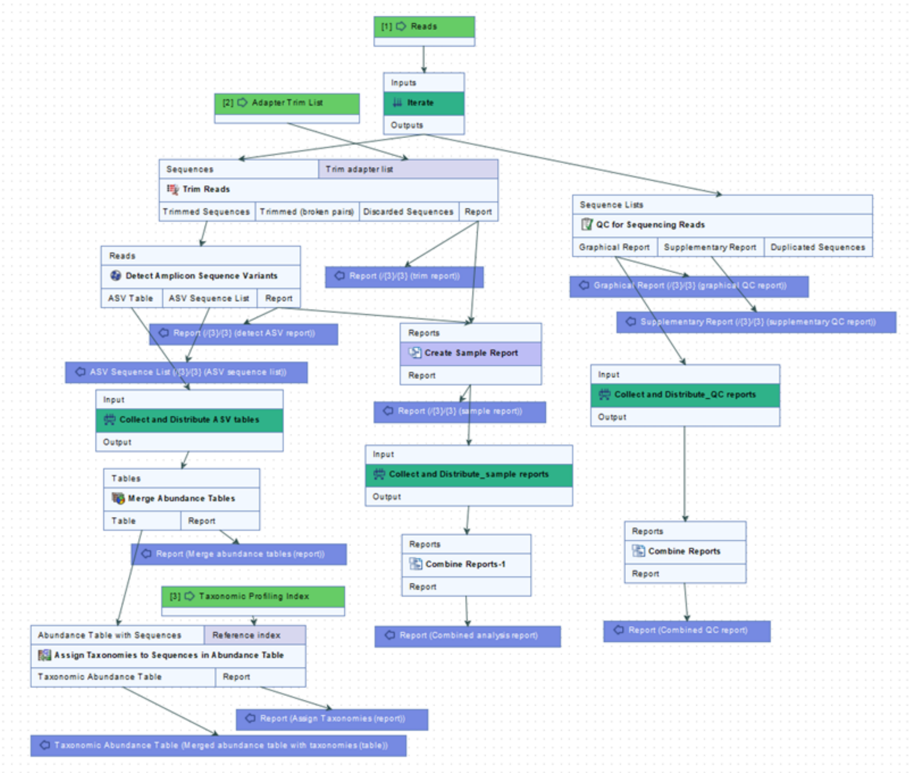

[1] Reads
Inputs
Iterate
Outputs
[2] Adapter Trim List
Sequences
Trim adapter list
Trim Reads
Trimmed Sequences
Trimmed (broken pairs)
Discarded Sequences
Report
Sequence Lists
QC for Sequencing Reads
Graphical Report
Supplementary Report
Duplicated Sequences
Reads
Detect Amplicon Sequence Variants
ASV Table
ASV Sequence List
Report
Report (/{3}/{3} (trim report))
Graphical Report (/{3}/{3} (graphical QC report))
Report (/{3}/{3} (detect ASV report))
Reports
Create Sample Report
Report
Supplementary Report (/{3}/{3} (supplementary QC report))
ASV Sequence List (/{3}/{3} (ASV sequence list))
Input
Collect and Distribute_QC reports
Output
Input
Collect and Distribute ASV tables
Output
Report (/{3}/{3} (sample report))
Input
Collect and Distribute_sample reports
Output
Tables
Merge Abundance Tables
Table
Report
Reports
Combine Reports
Report
Reports
Combine Reports-1
Report
Report (Merge abundance tables (report))
[3] Taxonomic Profiling Index
Report (Combined QC report)
Report (Combined analysis report)
Abundance Table with Sequences
Reference index
Assign Taxonomies to Sequences in Abundance Table
Taxonomic Abundance Table
Report
Report (Assign Taxonomies (report))
Taxonomic Abundance Table (Merged abundance table with taxonomies (table))

Figure 2.20: *Layout of the Detect Amplicon Sequence Variant and Assign Taxonomies workflow.*

Running this workflow will therefore give the following outputs: a phylogenetic tree of the OTUs, a diversity report for the alpha diversity and a PCoA for the beta diversity.

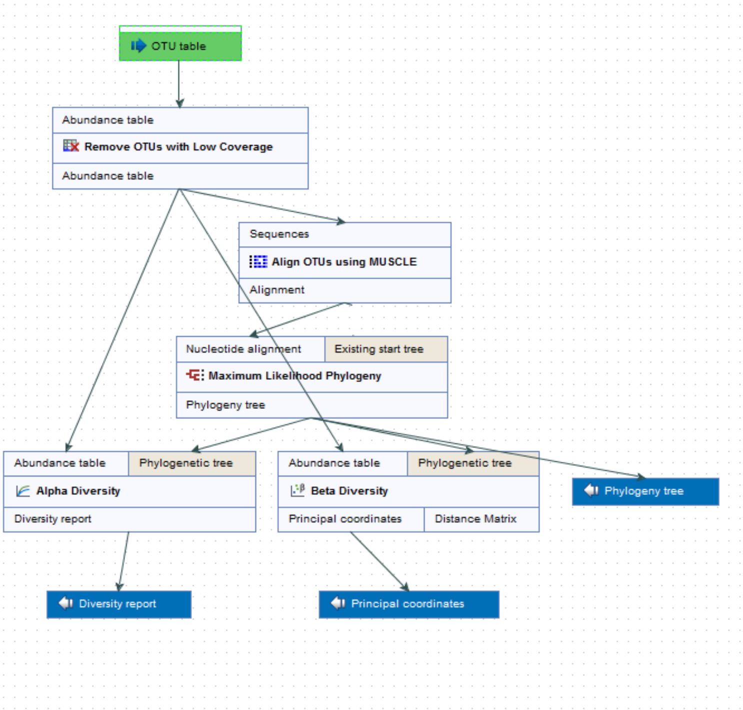


Figure 2.21: *Layout of the Alpha and Beta Diversities workflow.*

2.3 Typing and Epidemiology template workflows

The Typing and Epidemiology template workflows are found at:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( )

2.3.1 Compare Variants Across Samples

Compare Variants Across Samples can be used to compare samples originating from strains or species sharing a common reference. Input should be sequence lists of trimmed reads for which host reads have been removed, e.g. using **Taxonomic Profiling**, see section 5.2.

As the workflow removes duplicate mapped reads, amplicon data is not recommended as input. However, the workflow can be modified to work on amplicon data by opening a copy of the workflow, removing the **Remove Duplicate Mapped Reads** tool and saving the modified workflow.

To run the Compare Variants Across Samples workflow, go to

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( ) | **Compare Variants Across Samples** ( )

An overview of the workflow can be seen in figure 2.22.

- Select two or more read sets as input (figure 2.23). The workflow uses internal batching and creates an analysis for each sample as well as a combined variant track and SNP tree.

- Select the reference to use (figure 2.24). The reference should match all the samples

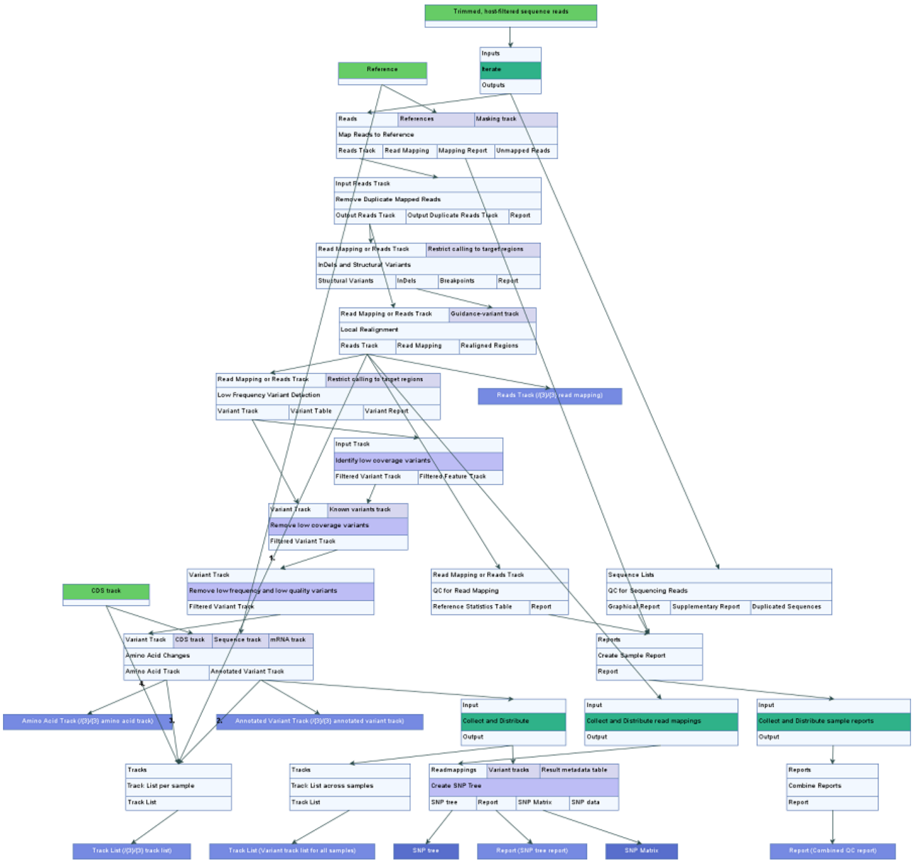
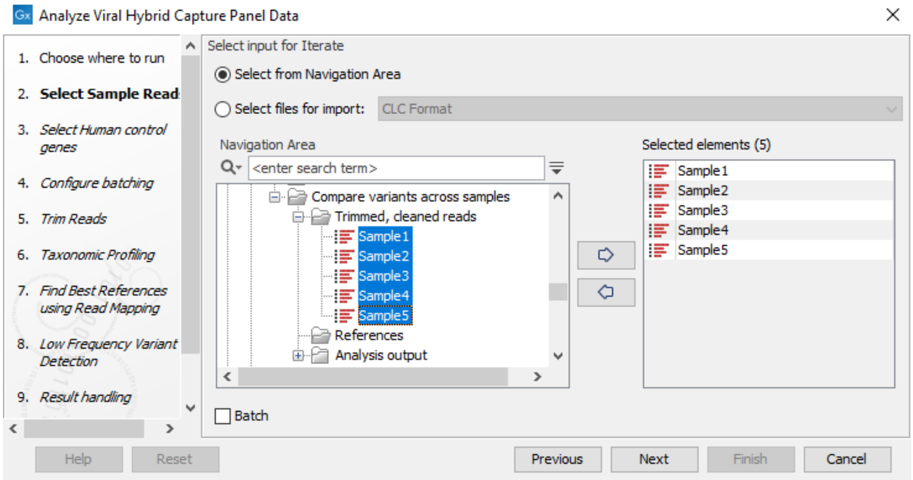


Figure 2.22: *An overview of the Compare Variants Across Samples workflow*


Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. **Select Sample Read**
3. *Select Human control genes*
4. *Configure batching*
5. *Trim Reads*
6. *Taxonomic Profiling*
7. *Find Best References using Read Mapping*
8. *Low Frequency Variant Detection*
9. *Result handling*

Select input for Iterate

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area

<enter search term>

Compare variants across samples
Trimmed, cleaned reads
Sample1
Sample2
Sample3
Sample4
Sample5
References
Analysis output

Selected elements (5)

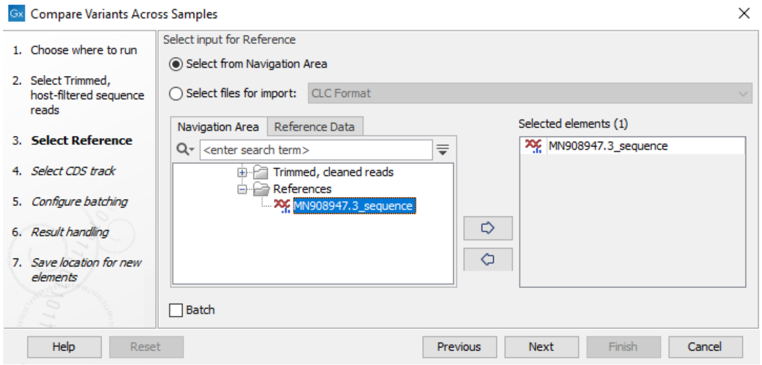
Sample1
Sample2
Sample3
Sample4
Sample5

□ Batch

Figure 2.23: *Select input data with common reference for analysis*

selected.

- Select a CDS track associated with the reference (figure 2.25).
- In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters.



Compare Variants Across Samples

1. Choose where to run
2. Select Trimmed, host-filtered sequence reads
3. **Select Reference**
4. *Select CDS track*
5. *Configure batching*
6. *Result handling*
7. *Save location for new elements*

Select input for Reference

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area | Reference Data

<enter search term>

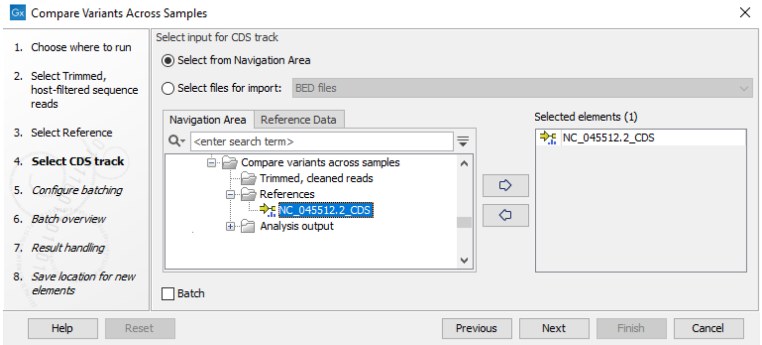
- Trimmed, cleaned reads
- References
  - MN908947.3_sequence

Selected elements (1)

MN908947.3_sequence

□ Batch

Figure 2.24: *Select reference*



Compare Variants Across Samples

1. Choose where to run
2. Select Trimmed, host-filtered sequence reads
3. Select Reference
4. **Select CDS track**
5. *Configure batching*
6. *Batch overview*
7. *Result handling*
8. *Save location for new elements*

Select input for CDS track

● Select from Navigation Area

○ Select files for import: BED files

Navigation Area | Reference Data

<enter search term>

- Compare variants across samples
  - Trimmed, cleaned reads
  - References
    - NC_045512.2_CDS
  - Analysis output

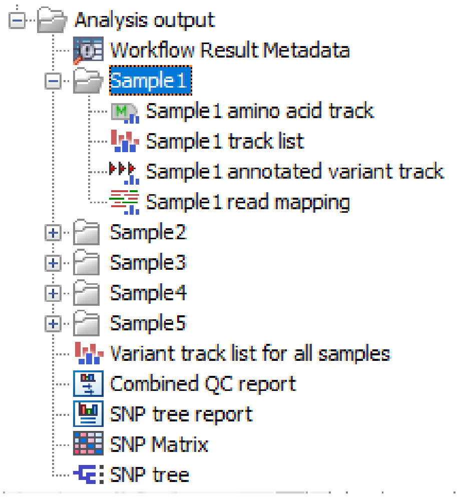
Selected elements (1)

NC_045512.2_CDS

□ Batch

Figure 2.25: *Select CDS track used to annotate variants*

Saving the workflow output will generate the files shown in (figure 2.26) and optionally, a workflow result metadata table. The workflow generates outputs for each batch analysis run as well as a folder for each sample. For each sample, the following is output:



- Analysis output
  - Workflow Result Metadata
  - Sample1
    - Sample1 amino acid track
    - Sample1 track list
    - Sample1 annotated variant track
    - Sample1 read mapping
  - Sample2
  - Sample3
  - Sample4
  - Sample5
  - Variant track list for all samples
  - Combined QC report
  - SNP tree report
  - SNP Matrix
  - SNP tree

Figure 2.26: *Output from Compare Variants Across Samples workflow*

- **Annotated variant track**: output from the **Low Frequency Variant Detection** tool after coverage and quality filtering. Note that it is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter. This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

- **Amino acid track**: amino acid track including amino acid changes resulting from the called variants.
- **Read mapping**: output from the **Local Realignment** tool, mapping of the reads to the specified reference. For increased sensitivity, duplicate mapped reads are removed before local realignment.
- **Track list**: output from the **Create Track List** tool. The track list combines the read mapping, variant, amino acid and CDS tracks. An example can be seen in figure 2.27.

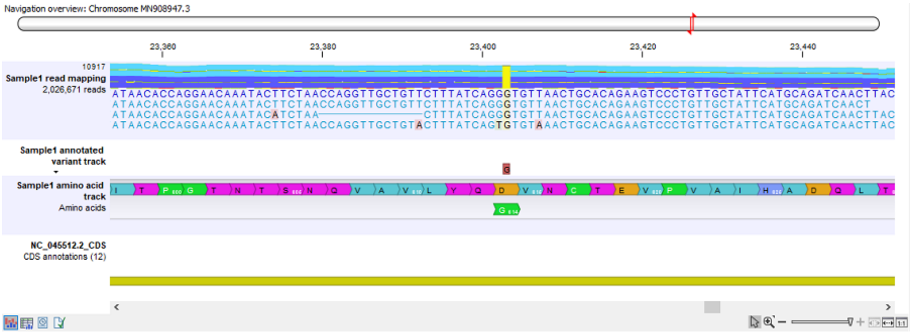
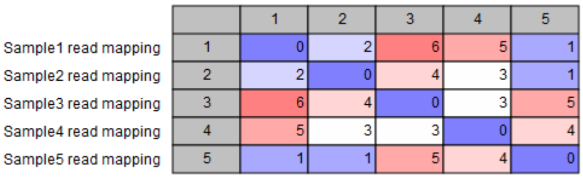


Figure 2.27: *The track list generated for each sample analysis*

For each batch analysis run, the following outputs are generated:

- **Variant track list for all samples**: output from the **Create Track List** tool. The track combines the variant tracks for all analyzed samples.
- **Combined QC report**: a combined report built from QC for sequencing reads, Read mapping summary and QC for read mapping. This report contains a summary of all analyzed samples.
- **SNP tree report**: summarizes the consequence of the applied filtering settings in the **Create SNP tree** tool, as well as a summary of ignored positions attributed to the different read mappings.
- **SNP matrix**: a matrix containing the pairwise number of SNP differences between all pairs of samples included in the analysis (see figure 2.28).



		1	2	3	4	5
Sample1 read mapping	1	0	2	6	5	1
Sample2 read mapping	2	2	0	4	3	1
Sample3 read mapping	3	6	4	0	3	5
Sample4 read mapping	4	5	3	3	0	4
Sample5 read mapping	5	1	1	5	4	0

Figure 2.28: *SNP matrix for pairwise comparisons of all samples included in the analysis*

- **SNP tree**: the output tree built from the SNPs called in all samples (see figure 2.29). A number of different visualizations are available, see section 9.1.2.

  Here, the leaf nodes have been colored according to geographic location of the collected samples.

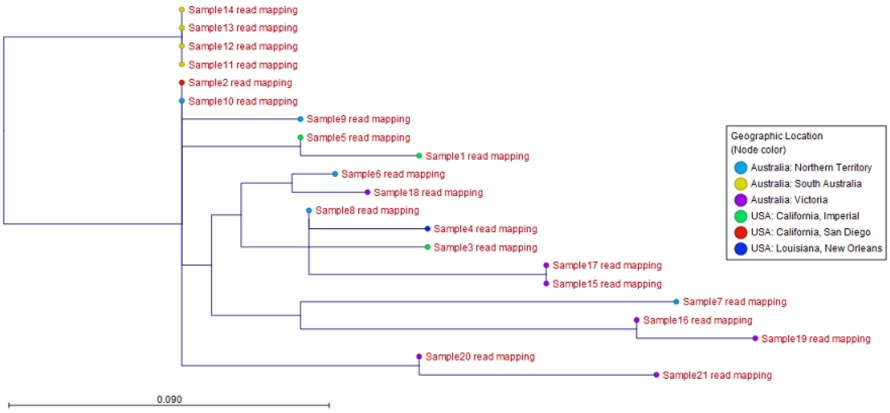


Figure 2.29: *An output SNP tree of 21 samples with leaf nodes colored using metadata annotation*

For more information on the Create SNP Tree tool, see section 9.1.

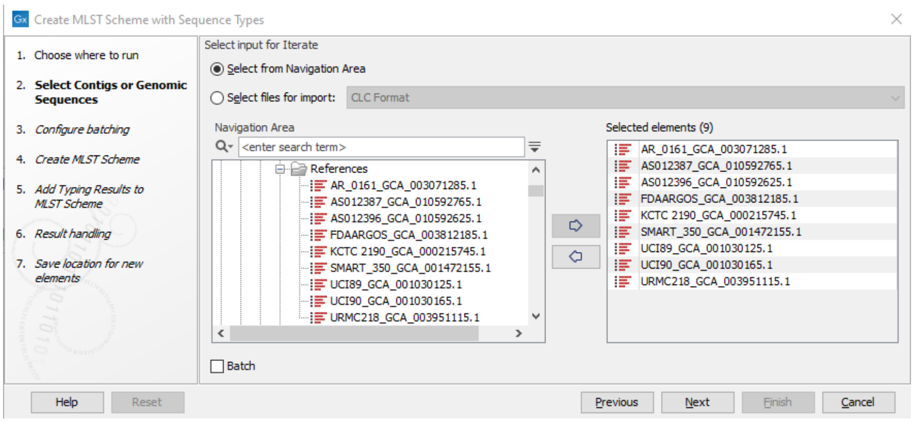
2.3.2 Create MLST Scheme with Sequence Types

The **Create MLST Scheme with Sequence Types** workflow creates a MLST scheme from references and adds sequence types by typing references and adding the results to the scheme.

To run the Create MLST Scheme with Sequence Types workflow, go to

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( ) | **Create MLST Scheme with Sequence Types** ( )

You can select one or more assemblies as input (figure 2.30). At least one of the assemblies must be annotated with CDS regions.


Create MLST Scheme with Sequence Types

1. Choose where to run
2. **Select Contigs or Genomic Sequences**
3. *Configure batching*
4. *Create MLST Scheme*
5. *Add Typing Results to MLST Scheme*
6. *Result handling*
7. *Save location for new elements*

Select input for Iterate

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area

<enter search term>

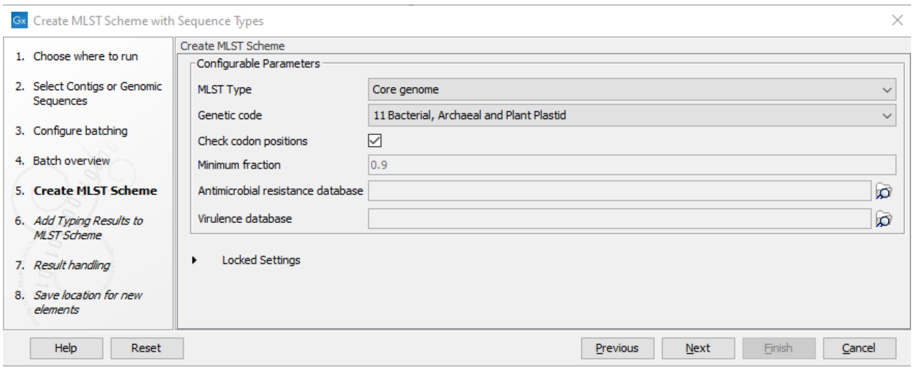
References
- AR_0161_GCA_003071285.1
- AS012387_GCA_010592765.1
- AS012396_GCA_010592625.1
- FDAARGOS_GCA_003812185.1
- KCTC 2190_GCA_000215745.1
- SMART_350_GCA_001472155.1
- UCI89_GCA_001030125.1
- UCI90_GCA_001030165.1
- URMC218_GCA_003951115.1

Selected elements (9)
- AR_0161_GCA_003071285.1
- AS012387_GCA_010592765.1
- AS012396_GCA_010592625.1
- FDAARGOS_GCA_003812185.1
- KCTC 2190_GCA_000215745.1
- SMART_350_GCA_001472155.1
- UCI89_GCA_001030125.1
- UCI90_GCA_001030165.1
- URMC218_GCA_003951115.1

☐ Batch

Figure 2.30: *Select the high-quality references serving as the basis for the scheme*

In "Create MLST scheme" dialog (figure 2.31), the settings for the scheme creation can be viewed and changed.



Create MLST Scheme with Sequence Types

1. Choose where to run
2. Select Contigs or Genomic Sequences
3. Configure batching
4. Batch overview
5. **Create MLST Scheme**
6. *Add Typing Results to MLST Scheme*
7. *Result handling*
8. *Save location for new elements*

Create MLST Scheme

Configurable Parameters

MLST Type: Core genome

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Check codon positions: ☑

Minimum fraction: 0.9

Antimicrobial resistance database:

Virulence database:

Locked Settings

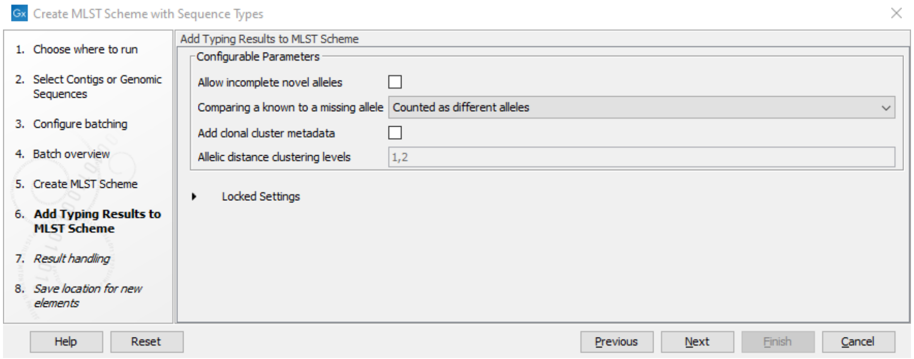
Figure 2.31: *Parameters for creating the initial scheme*

The parameters that can be set are:

- **MLST Type:** specifies the fraction of assemblies a locus must be present in to be included in the scheme. Options are: Core genome (corresponding to a fraction of 0.9), Whole genome (corresponding to a fraction of 0.1) or custom fraction.
- **Genetic code:** specifies the genetic code matching the input assemblies for a codon check.
- **Check codon positions:** if enabled, loci failing the specified codon check will not appear in the scheme. This should be disabled when working with organisms containing spliced genes.
- **Minimum fraction:** specifies the required fraction if custom fraction was selected in **MLST Type**.
- **Antimicrobial resistance database:** optional setting for specifying an antimicrobial resistance database to use for annotating loci in the scheme.
- **Virulence database:** optional setting for specifying a virulence database to use for annotating loci in the scheme.

In "Add Typing Results to MLST scheme" dialog (figure 2.32), sequence types will be added to the scheme. In addition, the following parameters can be specified:

- **Allow incomplete novel alleles:** whether only complete novel alleles (containing both start and stop codon) should be allowed. If incomplete novel alleles are not allowed, a sequence type with incomplete alleles for a locus will be added with missing alleles for that locus. If **Check codon positions** has been disabled (see figure 2.31), all alleles will be incomplete and consequently it will be necessary to allow adding incomplete alleles.
- **Comparing a known to a missing allele:** how to treat missing alleles when comparing a locus for a pair of sequence types.
- **Add clonal cluster metadata:** if selected, clonal cluster data will be added as metadata.



Create MLST Scheme with Sequence Types

1. Choose where to run
2. Select Contigs or Genomic Sequences
3. Configure batching
4. Batch overview
5. Create MLST Scheme
6. **Add Typing Results to MLST Scheme**
7. *Result handling*
8. *Save location for new elements*

Add Typing Results to MLST Scheme

Configurable Parameters

Allow incomplete novel alleles □

Comparing a known to a missing allele: Counted as different alleles

Add clonal cluster metadata □

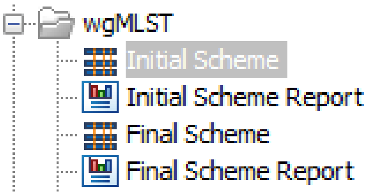
Allelic distance clustering levels: 1,2

Locked Settings

Figure 2.32: *Add Typing Results settings*

- **Allele distance clustering levels:** if clonal cluster data is added, specifies the allelic distance thresholds for adding clustering information.

In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters. Saving the output will generate the files shown in (figure 2.33) and optionally, a workflow result metadata table.



wgMLST
Initial Scheme
Initial Scheme Report
Final Scheme
Final Scheme Report

Figure 2.33: *The output from Create MLST Scheme with Sequence Types*

- **Initial Scheme Report:** the report from **Create MLST Scheme** tool.
- **Initial Scheme:** an empty scheme containing only loci.
- **Final Scheme Report:** the report from **Add Typing Results to MLST Scheme** tool.
- **Final Scheme:** the complete MLST Scheme containing loci and sequence types.

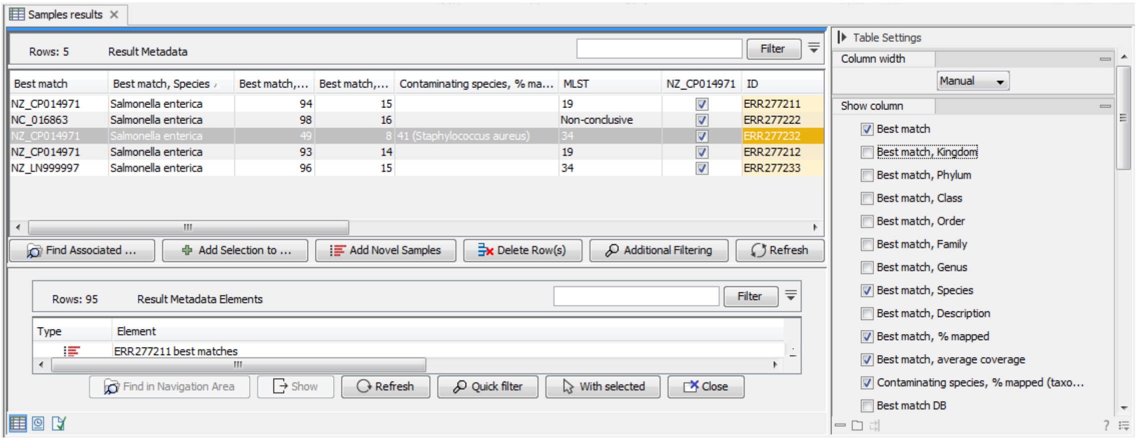
For more information on the tools and MLST schemes, see https://resources.qiagenbioinformatics.com/manuals/clcmgm/current/index.php?manual=MLST_Scheme_Tools.html

2.3.3 Map to Specified Reference

Once analysis has been performed using the **Type Among Multiple Species** workflow, the best matching reference is listed in the Result Metadata table (figure 2.34, see column Best match).

If all your samples share the same common reference, you can proceed to additional analyses without delay.

However there are cases where your samples have different Best match reference for a particular MLST scheme. And because creating a SNP Tree require a single common reference, you will



Rows: 5 Result Metadata

Best match	Best match, Species	Best match,...	Best match,...	Contaminating species, % ma...	MLST	NZ_CP014971	ID
NZ_CP014971	Salmonella enterica	94	15		19	☑	ERR277211
NC_016863	Salmonella enterica	98	16		Non-conclusive	☑	ERR277222
NZ_CP014971	Salmonella enterica	49	8	41 (Staphylococcus aureus)	34	☑	ERR277232
NZ_CP014971	Salmonella enterica	93	14		19	☑	ERR277212
NZ_LN999997	Salmonella enterica	96	15		34	☑	ERR277233

Rows: 95 Result Metadata Elements

Type	Element
	ERR277211 best matches

Table Settings

Column width: Manual

Show column

- ☑ Best match
- □ Best match, Kingdom
- □ Best match, Phylum
- □ Best match, Class
- □ Best match, Order
- □ Best match, Family
- □ Best match, Genus
- ☑ Best match, Species
- □ Best match, Description
- ☑ Best match, % mapped
- ☑ Best match, average coverage
- ☑ Contaminating species, % mapped (taxo...
- □ Best match DB

Figure 2.34: *Best match references are listed for each row in the Result Metadata Table.*

need to identify the best matching common reference for all your samples using a K-mer Tree, as well as subsequently re-map your samples to this common reference.

If you already know the common reference for the sample you want to use to create a SNP tree, you can directly specify that reference in the re-map workflow. Otherwise, finding a common reference is described in more details in section 8.1.1.

In short, **to identify a common reference across multiple clades within the Result Metadata Table:**

- **Select** samples to which a common best matching references should be identified.
- **Click** on the **Find Associated Data** ( ) button to find their associated Metadata Elements.
- **Click** on the **Quick Filtering** ( ) button and select the option **Filter for K-mer Tree** to find Metadata Elements with the Role = Trimmed Reads.
- **Select** the relevant Metadata Element files.
- **Click** on the **With selected** ( ) button.
- **Select** the **Create K-mer Tree** action and follow the wizard as described in section 9.2.

The common reference, chosen as sharing the closest common ancestor with the clade of isolates under study in the k-mer tree, is subsequently used as a reference for the **Map to Specified Reference** workflow (figure 2.35) that will perform a re-mapping of the reads followed by variant calling.

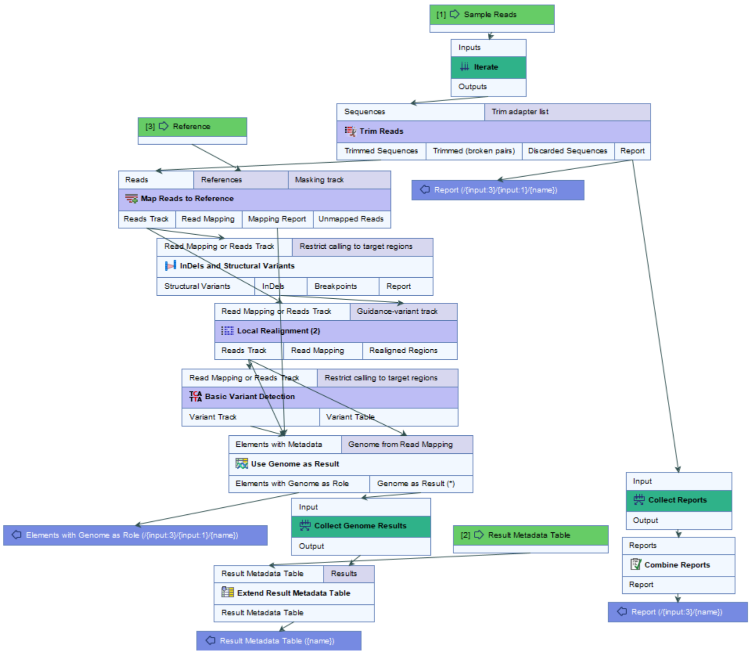


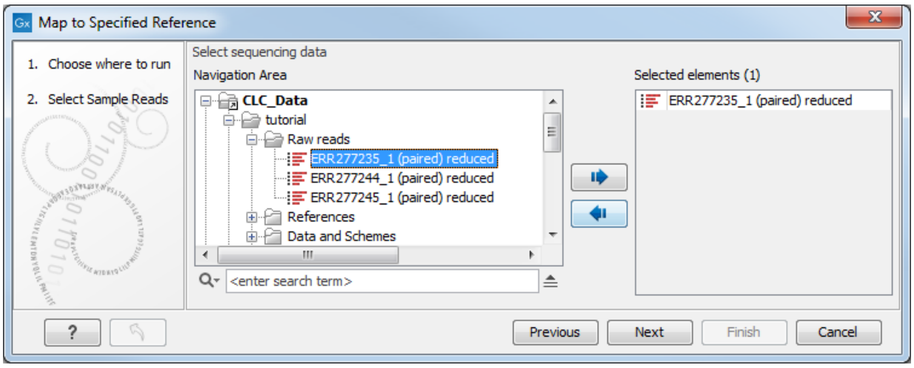
Figure 2.35: *Overview of the template Map to Specified Reference workflow.*

How to run the Map to Specified Reference workflow

This workflow is intended for read mapping and variant calling of the samples to a common reference. To run the workflow, go to:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( ) | **Map to Specified Reference** ( )

1. Specify the sample(s) or folder(s) of samples you would like to type (figure 2.36) and click **Next**. Remember that if you select several items, they will be run as batch units.


Map to Specified Reference

1. Choose where to run
2. Select Sample Reads

Select sequencing data

Navigation Area

- CLC_Data
  - tutorial
    - Raw reads
      - ERR277235_1 (paired) reduced
      - ERR277244_1 (paired) reduced
      - ERR277245_1 (paired) reduced
    - References
    - Data and Schemes

<enter search term>

Selected elements (1)

ERR277235_1 (paired) reduced

Figure 2.36: *Select the reads from the sample(s) you would like to type.*

2. Specify the **Result Metadata Table** you want to use (figure 2.37) and click **Next**.

3. Select the reference you obtained from the previous workflows - provided that it was the same reference for all the samples you want to re-map - or determined earlier from your K-mer tree if the samples you want to re-map had different best match references. Click **Next**.

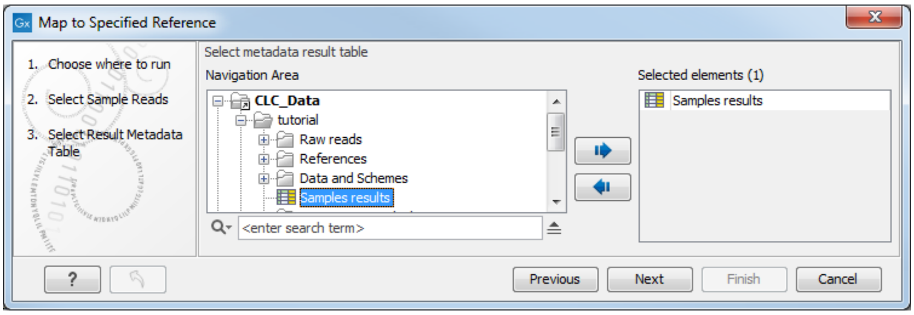


Figure 2.37: *Select the metadata table you would like to use.*

4. Define batch units using organisation of input data to create one run per input or use a metadata table to define batch units. Click **Next**.

5. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 2.38).

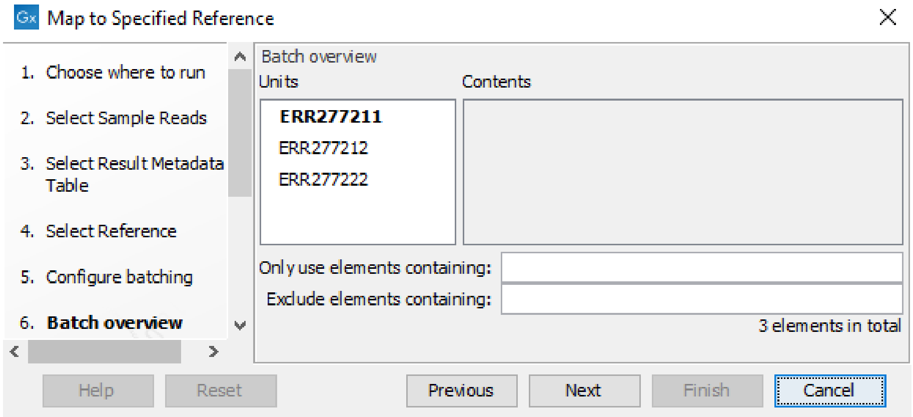


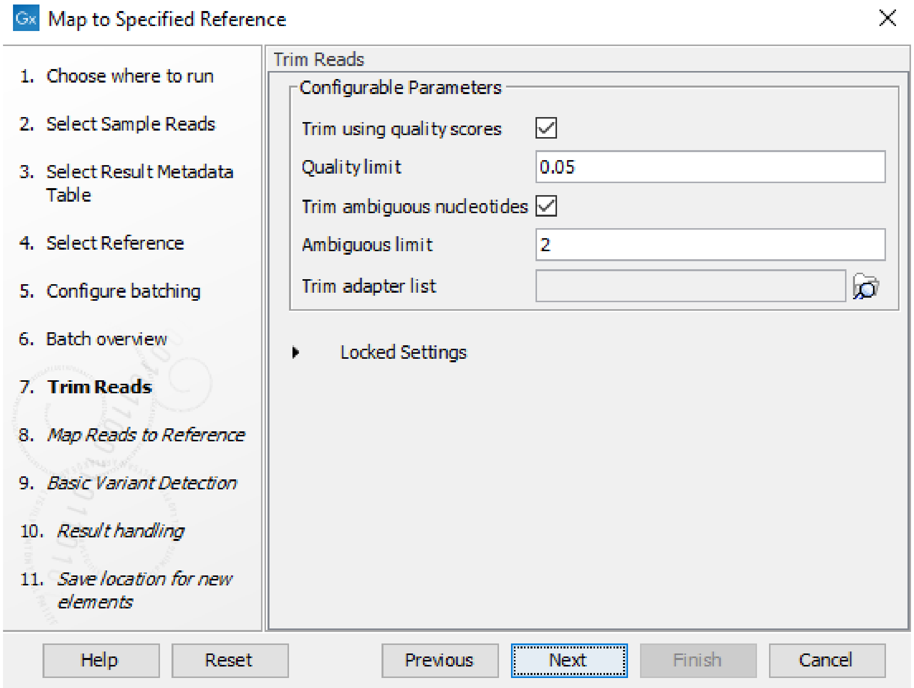
Figure 2.38: *Choose which of the samples present in the selected folder(s) you want to analyze.*

6. You can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 2.39). Learn about trim adapter lists at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_adapter_list.html.

   The parameters that can be set are:

   - **Ambiguous trim**: if checked, this option trims the sequence ends based on the presence of ambiguous nucleotides (typically N).
   - **Ambiguous limit**: defines the maximal number of ambiguous nucleotides allowed in the sequence after trimming.
   - **Quality trim**: if checked, and if the sequence files contain quality scores from a base-caller algorithm, this information can be used for trimming sequence ends.
   - **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.

   Click **Next**.



Trim Reads

Configurable Parameters

Trim using quality scores ☑

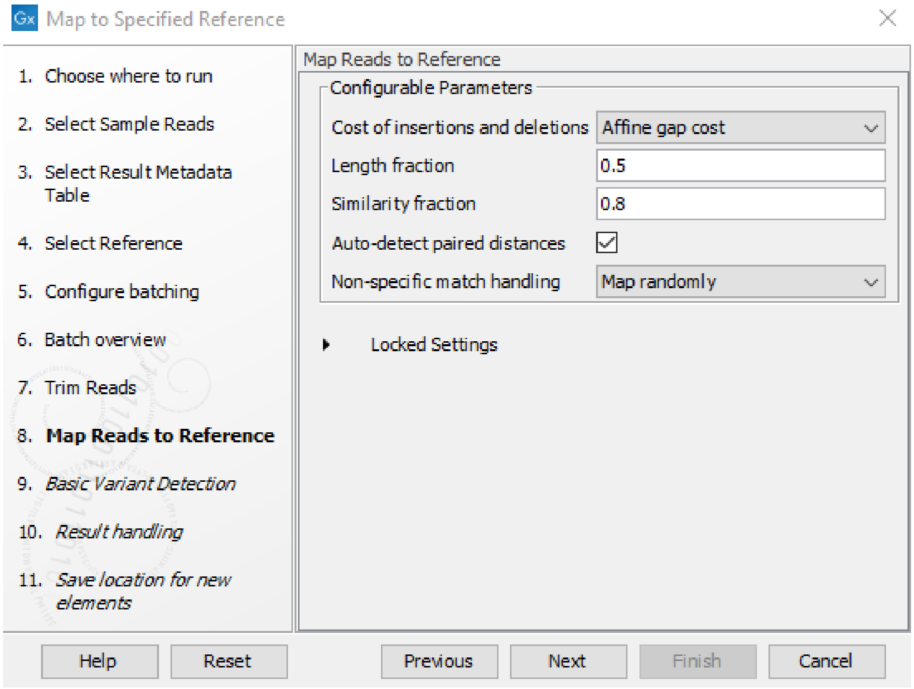
Quality limit 0.05

Trim ambiguous nucleotides ☑

Ambiguous limit 2

Trim adapter list

Figure 2.39: *You can choose to trim adapter sequences from your sequencing reads.*



Map Reads to Reference

Configurable Parameters

Cost of insertions and deletions Affine gap cost

Length fraction 0.5

Similarity fraction 0.8

Auto-detect paired distances ☑

Non-specific match handling Map randomly

Figure 2.40: *Specify the parameters for the Maps Reads to Reference tool.*

7. Specify the parameters for the **Maps Reads to Reference** tool (figure 2.40).

   The parameters that can be set are:

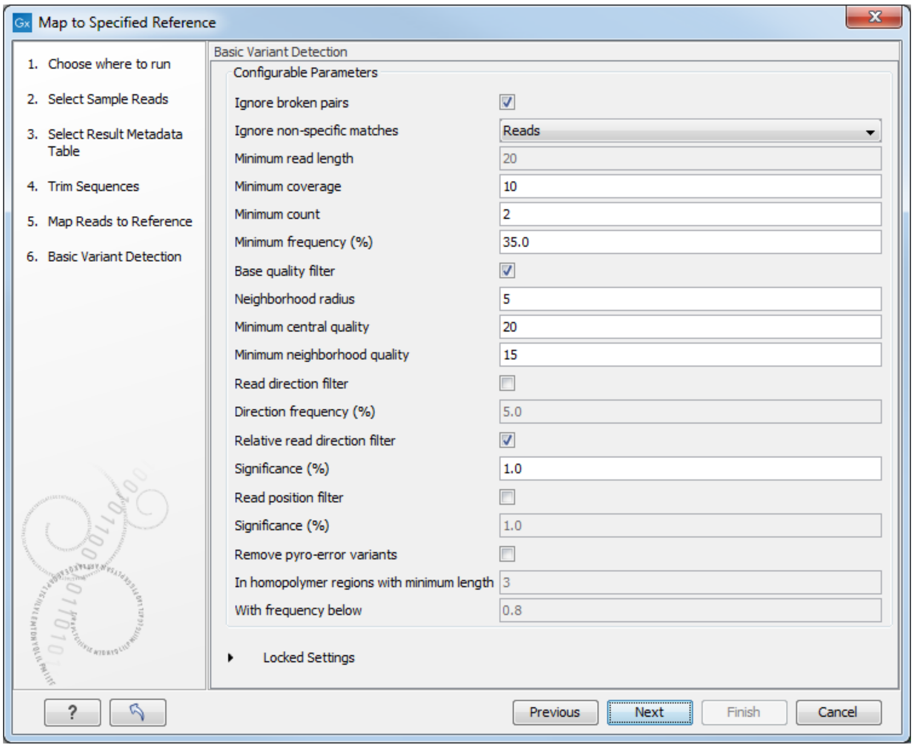
   - **Cost of insertion and deletions**: You can choose affine or linear gap cost.
   - **Length fraction**: The minimum percentage of the total alignment length that must match the reference sequence at the selected similarity fraction. A fraction of 0.5 means that at least half of the alignment must 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
- **Auto-detect paired sequences**: This will determine the paired distance (insert size) of paired data sets. If several paired sequence lists are used as input, a separate calculation is done for each one to allow for different libraries in the same run.
- **Non-specific match handling**: You can choose from the drop down menu whether you would like to ignore or map randomly the non specific matches.

Click **Next**.

8. Specify the parameters for the **Basic Variant Detection** tool (figure 2.41) before clicking **Next**.



1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Trim Sequences
5. Map Reads to Reference
6. Basic Variant Detection

Ignore broken pairs ☑
Ignore non-specific matches: Reads
Minimum read length: 20
Minimum coverage: 10
Minimum count: 2
Minimum frequency (%): 35.0
Base quality filter ☑
Neighborhood radius: 5
Minimum central quality: 20
Minimum neighborhood quality: 15
Read direction filter ☐
Direction frequency (%): 5.0
Relative read direction filter ☑
Significance (%): 1.0
Read position filter ☐
Significance (%): 1.0
Remove pyro-error variants ☐
In homopolymer regions with minimum length: 3
With frequency below: 0.8

Figure 2.41: *Specify the parameters to be used for the Basic Variant Detection tool.*

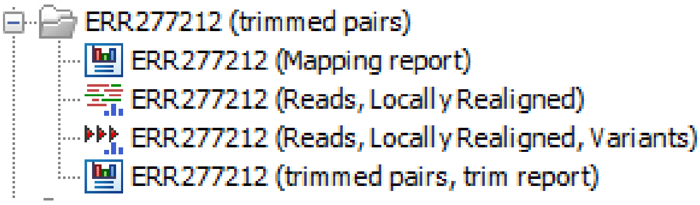
The parameters that can be set are:

- **Ignore broken pairs**: You can choose to ignore broken pairs by clicking this option.
- **Ignore non-specific matches**: You can choose to ignore non-specific matches between reads, regions or to not ignore them at all.
- **Minimum read length**: Only variants in reads longer than this size are called.
- **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.
- **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.
- **Neighborhood radius**: Determine how far away from the current variant the quality assessment should extend.
- **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.
- **Read direction filters**: The read direction filter removes variants that are almost exclusively present in either forward or reverse reads.
- **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.
- **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**: 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.
- **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.
- **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 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.

9. In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters. Choose to save the results and click on the button labeled **Finish**.

Four outputs are generated per input sample (figure 2.42):



ERR277212 (trimmed pairs)
- ERR277212 (Mapping report)
- ERR277212 (Reads, Locally Realigned)
- ERR277212 (Reads, Locally Realigned, Variants)
- ERR277212 (trimmed pairs, trim report)

Figure 2.42: *Output files from the Map to Specified Reference workflow.*

- **Mapping Summary report**: summary report about the mapping process, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Summary_mapping_report.html.
- **Trim report**: summary report for the trimming, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html.
- **Reads Track**: output from the **Local Realignment** tool
- **Variant Track**: output from the **Basic Variant Detection** tool. Note that it is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter. This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

You now have the data necessary to create a SNP tree for your samples as explained in section 9.1.

Note that the tool will output, among other files, variant tracks. It is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter.This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

2.3.4 Type Among Multiple Species

The **Type Among Multiple Species** workflow is designed for typing a sample among multiple predefined species (figure 2.43).

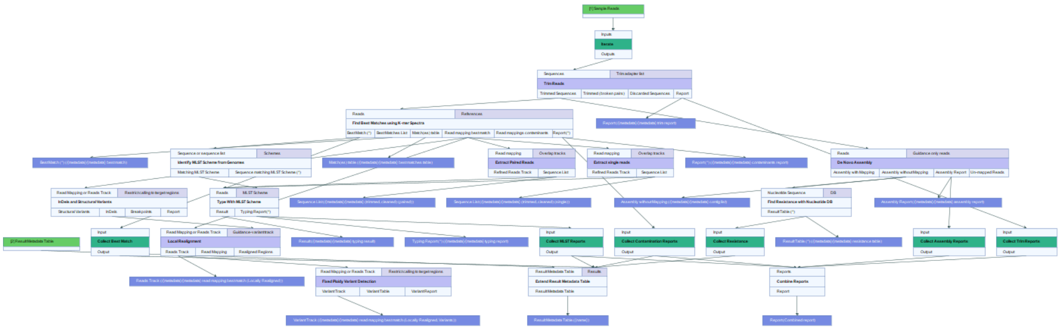


Figure 2.43: *Overview of the template Type Among Multiple Species workflow.*

It allows identification of the closest matching reference species among the user specified reference list(s) which may represent multiple species. The workflow identifies the associated MLST scheme and type, determines variants found when mapping the sample data against the identified best matching reference, and finds occurring resistance genes if they match genes within the user specified resistance database.

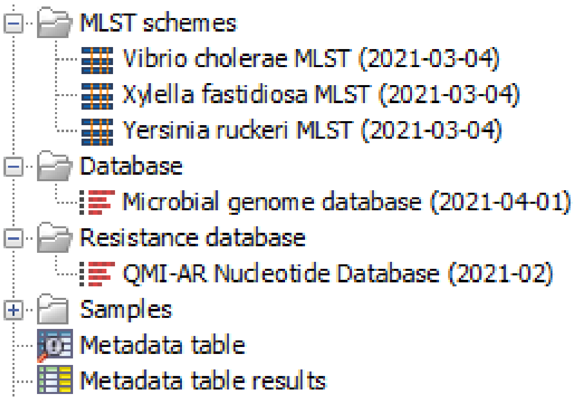
The workflow also automatically associates the analysis results to the user specified Result Metadata Table. For details about searching and quick filtering among the sample metadata and generated analysis result data (see section 19.2.2).

Preliminary steps to run the Type Among Multiple Species workflow

Before starting the workflow,

- Download microbial genomes using either the **Download Custom Microbial Reference Database** tool, the prokaryotic databases from the **Download Curated Microbial Reference Database** tool or the **Download Pathogen Reference Database** tool (see section VI). Databases can also be created using the **Update Sequence Attributes in Lists** tool.
- Download the MLST schemes using the **Download MLST Scheme** tool (see section 13.2).
- Download the database for the **Find Resistance with Nucleotide Database** tool using the **Download Resistance Database** tool (see section 17.1).
- Create a New Result Metadata table using the **Create Result Metadata Table** tool (see section 19.2.1).

When you are ready to start the workflows, your navigation area should look similar to the figure 2.44.



MLST schemes
Vibrio cholerae MLST (2021-03-04)
Xylella fastidiosa MLST (2021-03-04)
Yersinia ruckeri MLST (2021-03-04)
Database
Microbial genome database (2021-04-01)
Resistance database
QMI-AR Nucleotide Database (2021-02)
Samples
Metadata table
Metadata table results

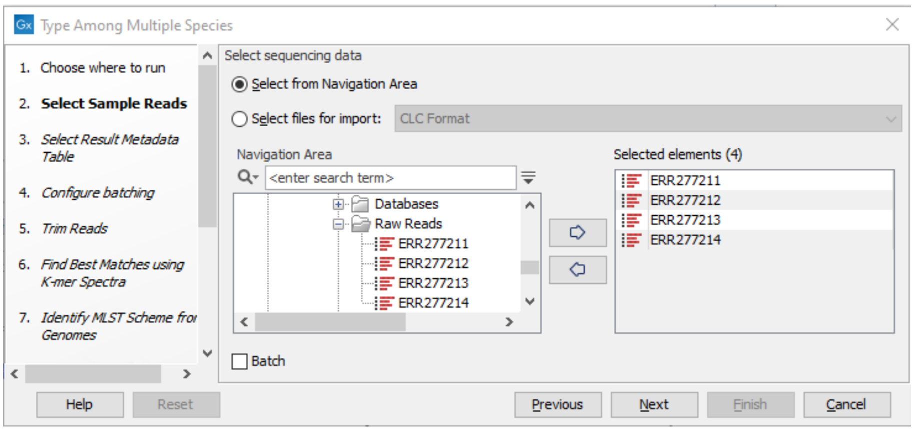
Figure 2.44: *Overview of the Navigation area after creating the result metadata table and downloading the databases and MLST schemes necessary to run the workflows.*

How to run the Type Among Multiple Species workflow

To run the workflow for one or more samples containing multiple species, go to

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( ) | **Type Among Multiple Species** ( )

1. Specify the **sample(s)** or folder(s) of samples you would like to type (figure 2.45) and click **Next**. Remember that if you select several items, they will be run as batch units.
2. Specify the **Result Metadata Table** you want to use (figure 2.46) and click **Next**.



Type Among Multiple Species

1. Choose where to run
2. **Select Sample Reads**
3. *Select Result Metadata Table*
4. *Configure batching*
5. *Trim Reads*
6. *Find Best Matches using K-mer Spectra*
7. *Identify MLST Scheme fro Genomes*

Select sequencing data

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area

<enter search term>

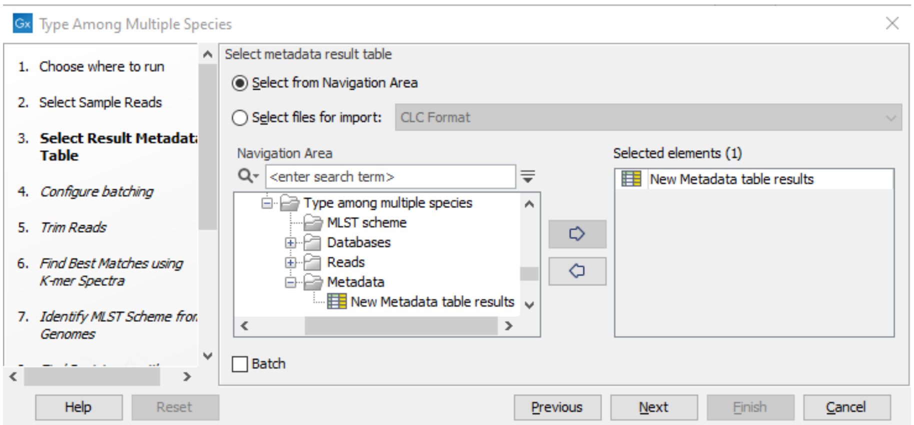
- Databases
- Raw Reads
  - ERR277211
  - ERR277212
  - ERR277213
  - ERR277214

Selected elements (4)

- ERR277211
- ERR277212
- ERR277213
- ERR277214

□ Batch

Figure 2.45: *Select the reads from the sample(s) you would like to type.*



Type Among Multiple Species

1. Choose where to run
2. Select Sample Reads
3. **Select Result Metadat Table**
4. *Configure batching*
5. *Trim Reads*
6. *Find Best Matches using K-mer Spectra*
7. *Identify MLST Scheme fro Genomes*

Select metadata result table

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area

<enter search term>

- Type among multiple species
  - MLST scheme
  - Databases
  - Reads
  - Metadata
    - New Metadata table results

Selected elements (1)

- New Metadata table results

□ Batch

Figure 2.46: *Select the metadata table you would like to use.*

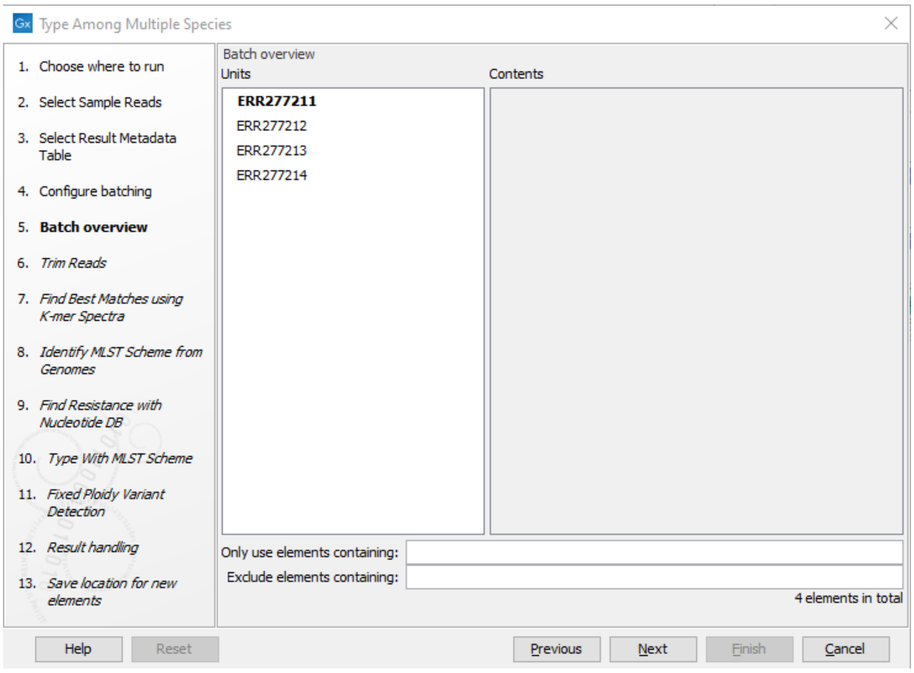
3. Define batch units using organisation of input data to create one run per input or use a metadata table to define batch units. Click **Next**.

4. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 2.47).

5. You can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 2.48). Learn about trim adapter lists at `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_adapter_list.html`.

   The parameters that can be set are:

   - **Trim ambiguous nucleotides**: if checked, this option trims the sequence ends based on the presence of ambiguous nucleotides (typically N).
   - **Maximum number of ambiguities**: defines the maximal number of ambiguous nucleotides allowed in the sequence after trimming.
   - **Trim using quality scores**: if checked, and if the sequence files contain quality scores from a base-caller algorithm, this information can be used for trimming sequence ends.


Type Among Multiple Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. Type With MLST Scheme
11. Fixed Ploidy Variant Detection
12. Result handling
13. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Batch overview

Units: ERR277211, ERR277212, ERR277213, ERR277214

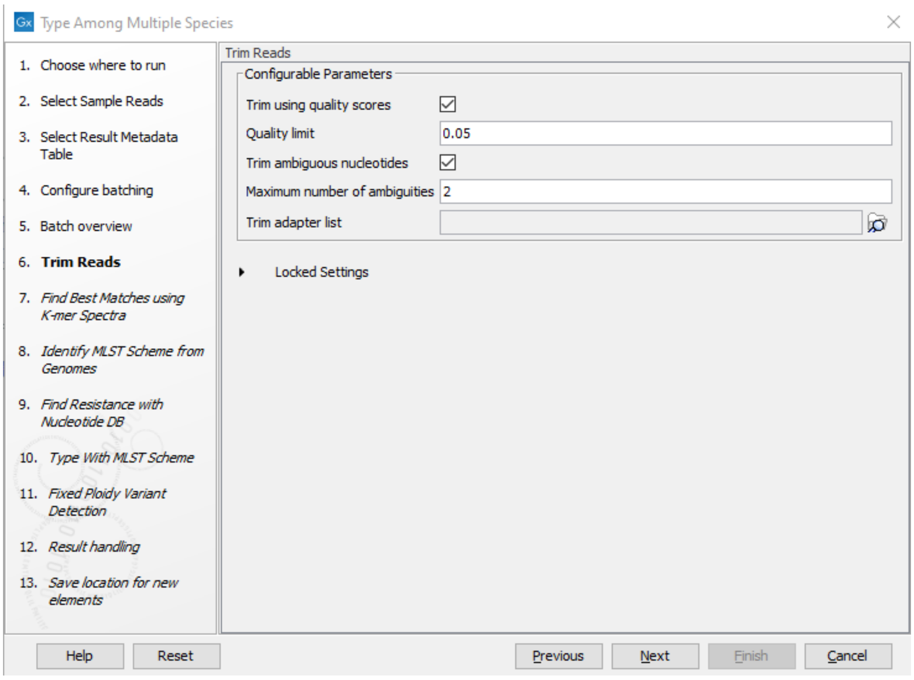
Contents

Only use elements containing:

Exclude elements containing:

4 elements in total

Figure 2.47: *Choose which of the samples present in the selected folder(s) you want to analyze.*


Type Among Multiple Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. Type With MLST Scheme
11. Fixed Ploidy Variant Detection
12. Result handling
13. Save location for new elements
Locked Settings
Help
Reset
Previous
Next
Finish
Cancel

Trim Reads

Configurable Parameters

Trim using quality scores ☑

Quality limit 0.05

Trim ambiguous nucleotides ☑

Maximum number of ambiguities 2

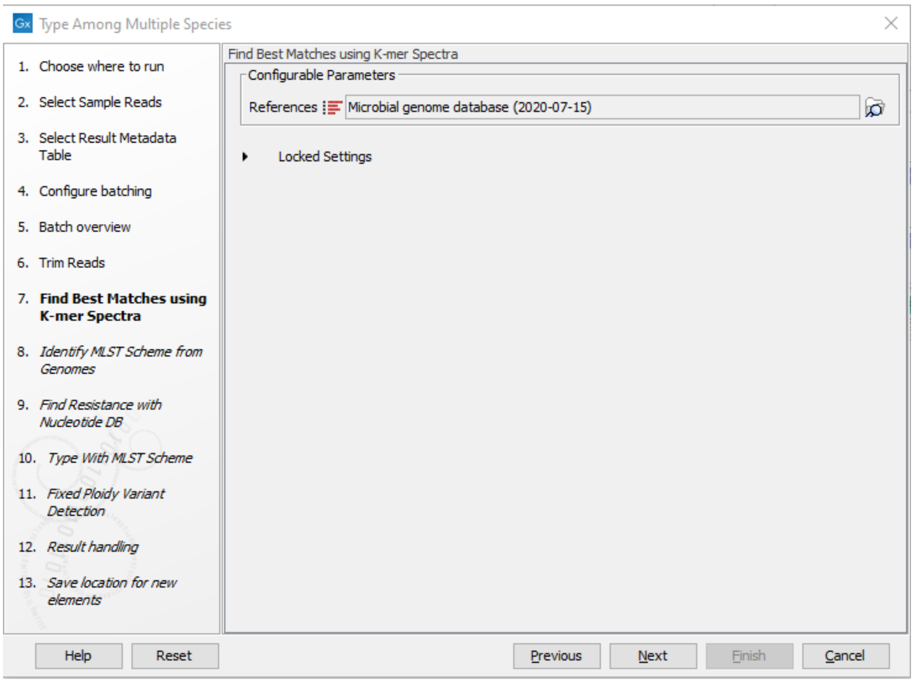
Trim adapter list

Figure 2.48: *You can choose to trim adapter sequences from your sequencing reads.*

- **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.

Click **Next**.

6. Choose the **species-specific references** to be used by the **Find Best Matches using K-mer Spectra** tool (figure 2.49). The list can be a fully customized list, the downloaded bacterial genomes from NCBI list (see section 15.1.1) or a subset of it. Click **Next**.


Type Among Multiple Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. Type With MLST Scheme
11. Fixed Ploidy Variant Detection
12. Result handling
13. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Find Best Matches using K-mer Spectra

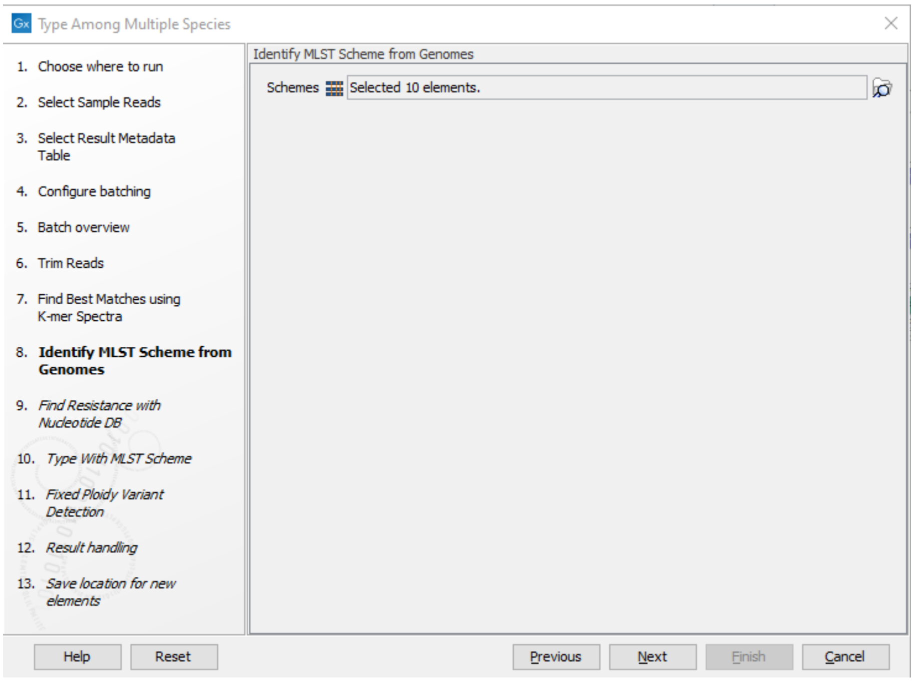
Configurable Parameters

References: Microbial genome database (2020-07-15)

Locked Settings

Figure 2.49: *Specify the references for the Find Best Matches using K-mer Spectra tool.*

7. Specify **MLST schemes** to be used for the **Identify MLST Scheme from Genomes** tool so they correspond to corresponding to the chosen reference list(s) (figure 2.50).


Type Among Multiple Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. Type With MLST Scheme
11. Fixed Ploidy Variant Detection
12. Result handling
13. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Identify MLST Scheme from Genomes

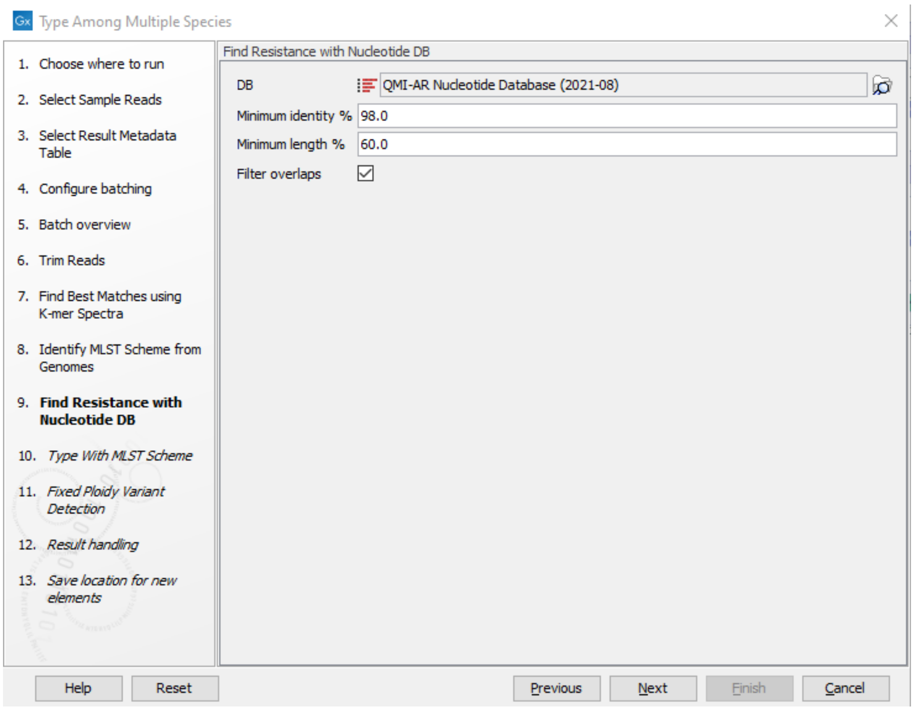
Schemes: Selected 10 elements.

Figure 2.50: *Specify the schemes that best describe your sample(s).*

8. Specify the **resistance database** (figure 2.51) and set the parameters for the **Find Resistance with Nucleotide Database** tool.

   The parameters that can be set are:

   - **Minimum Identity %**: is the threshold for the minimum percentage of nucleotides that



Type Among Multiple Species

1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. **Find Resistance with Nucleotide DB**
10. *Type With MLST Scheme*
11. *Fixed Ploidy Variant Detection*
12. *Result handling*
13. *Save location for new elements*

Find Resistance with Nucleotide DB

DB: QMI-AR Nucleotide Database (2021-08)

Minimum identity %: 98.0

Minimum length %: 60.0

Filter overlaps: ☑

Figure 2.51: *Specify the resistance database to be used for the Find Resistance with Nucleotide Database tool.*

are identical between the best matching resistance gene in the database and the corresponding sequence in the genome.

- **Minimum Length %**: reflect the percentage of the total resistance gene length that a sequence must overlap a resistance gene to count as a hit for that gene. Here represented as a percentage of the total resistance gene length.
- **Filter overlaps**: will perform extra filtering of results per contig, where one hit is contained by the other with a preference for the hit with the higher number of aligned nucleotides (length * identity).

Click **Next**.

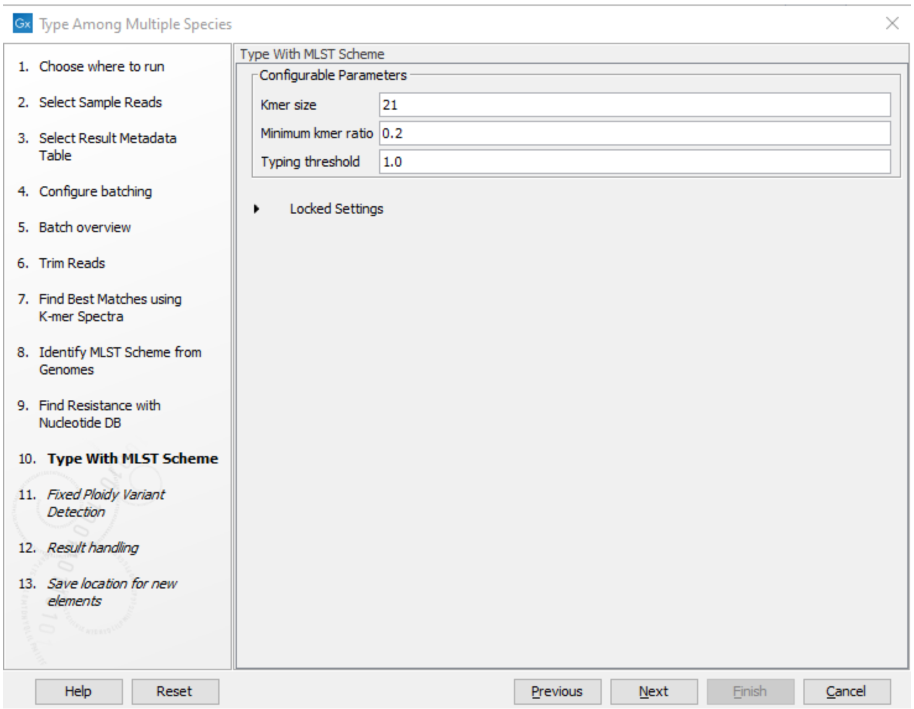
9. Specify the parameters for the **Type with MLST Scheme** tool (figure 2.52).

   The parameters that can be set are:

   - **Kmer size**: determines the number of nucleotides in the kmer - raising this setting might increase specificity at the cost of some sensitivity.
   - **Typing threshold**: determines how many of the kmers in a sequence type that needs to be identified before a typing is considered conclusive. The default setting of 1.0 means that all kmers in all alleles must be matched.
   - **Minimum kmer ratio**: the minimum kmer ratio of the least occurring kmer and the average kmer hit count. If an allele scores higher than this threshold it is classified as a high-confidence call.

   Click **Next**.

10. Specify the parameters for the **Fixed Ploidy Variant Detection** tool (figure 2.53) before clicking **Next**.



Type Among Multiple Species

1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. **Type With MLST Scheme**
11. *Fixed Ploidy Variant Detection*
12. *Result handling*
13. *Save location for new elements*

Type With MLST Scheme

Configurable Parameters

Kmer size: 21

Minimum kmer ratio: 0.2

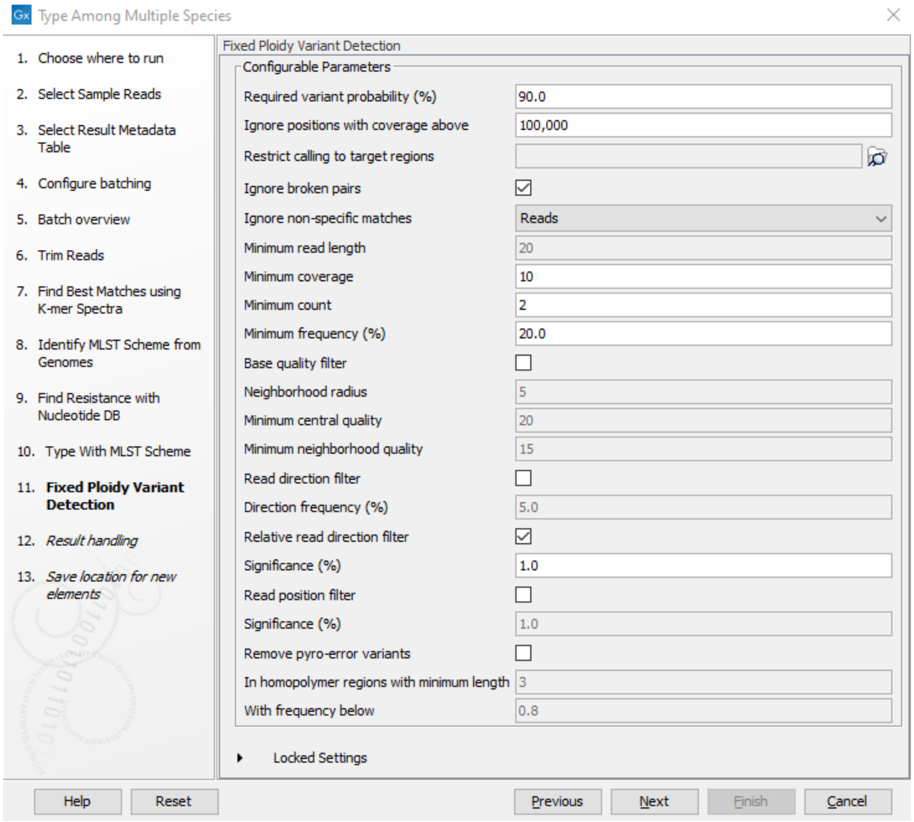
Typing threshold: 1.0

Locked Settings

Figure 2.52: *Specify the parameters for MLST typing.*

The parameters that can be set are:

- **Required variant probability (%)**: 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.
- **Ignore positions with coverage above**: Ignore positions with a read-coverage larger than this value.
- **Restrict calling to target regions**: Select a region track to specify the regions in which variants should be called.
- **Ignore broken pairs**: You can choose to ignore broken pairs by clicking this option.
- **Ignore non-specific matches**: You can choose to ignore non-specific matches between reads, regions or to not ignore them at all.
- **Minimum read length**: Only variants in reads longer than this size are called.
- **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.
- **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.


Type Among Multiple Species

1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Find Best Matches using K-mer Spectra
8. Identify MLST Scheme from Genomes
9. Find Resistance with Nucleotide DB
10. Type With MLST Scheme
11. **Fixed Ploidy Variant Detection**
12. *Result handling*
13. *Save location for new elements*

Fixed Ploidy Variant Detection

Configurable Parameters

Required variant probability (%)	90.0
Ignore positions with coverage above	100,000
Restrict calling to target regions	
Ignore broken pairs	☑
Ignore non-specific matches	Reads
Minimum read length	20
Minimum coverage	10
Minimum count	2
Minimum frequency (%)	20.0
Base quality filter	□
Neighborhood radius	5
Minimum central quality	20
Minimum neighborhood quality	15
Read direction filter	□
Direction frequency (%)	5.0
Relative read direction filter	☑
Significance (%)	1.0
Read position filter	□
Significance (%)	1.0
Remove pyro-error variants	□
In homopolymer regions with minimum length	3
With frequency below	0.8

▸ Locked Settings

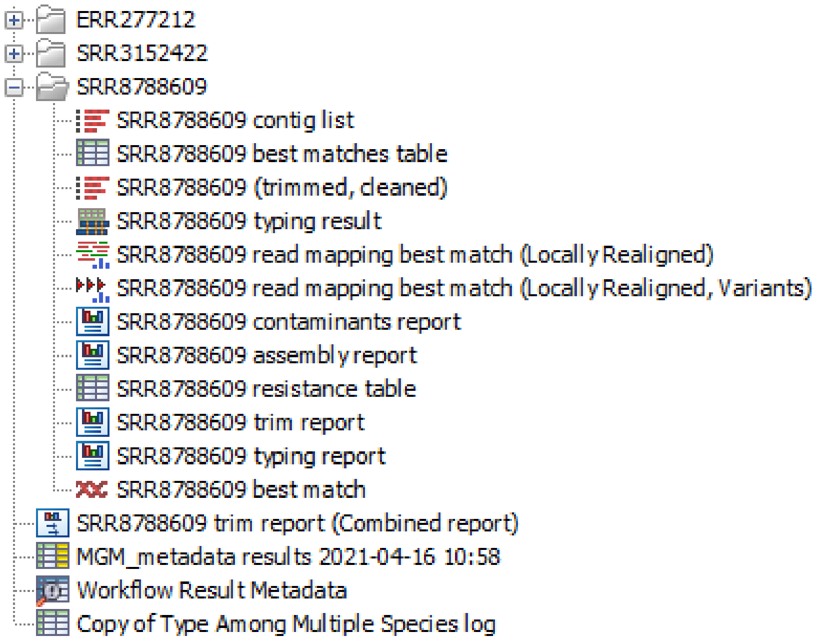
Figure 2.53: *Specify the parameters to be used for the Fixed Ploidy Variant Detection tool.*

- **Neighborhood radius**: Determine how far away from the current variant the quality assessment should extend.
- **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.
- **Read direction filters**: The read direction filter removes variants that are almost exclusively present in either forward or reverse reads.
- **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.
- **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**: 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.

- **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.
- **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 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.

11. In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters. Choose to save the results (we recommend to create a new folder to this effect) and click on the button labeled **Finish**.

Outputs are generated on a per sample basis and on a summary level. You can find them all in the new folder you created to save them (figure 2.54), but those marked with a (*) have also been added automatically to the New Metadata Result Table (see section 2.3.5 to understand where your results have been saved).



- ERR277212
- SRR3152422
- SRR8788609
  - SRR8788609 contig list
  - SRR8788609 best matches table
  - SRR8788609 (trimmed, cleaned)
  - SRR8788609 typing result
  - SRR8788609 read mapping best match (Locally Realigned)
  - SRR8788609 read mapping best match (Locally Realigned, Variants)
  - SRR8788609 contaminants report
  - SRR8788609 assembly report
  - SRR8788609 resistance table
  - SRR8788609 trim report
  - SRR8788609 typing report
  - SRR8788609 best match
- SRR8788609 trim report (Combined report)
- MGM_metadata results 2021-04-16 10:58
- Workflow Result Metadata
- Copy of Type Among Multiple Species log

Figure 2.54: *Output files from the Type Among Multiple Species workflow.*

For each sample, the following outputs are generated:

- **Trim report**: report from the **Trim Sequences** tool (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html).
- (*)**Contaminants report**: lists the best match as well as possible contaminants along with coverage level distributions for each reference genome listed.

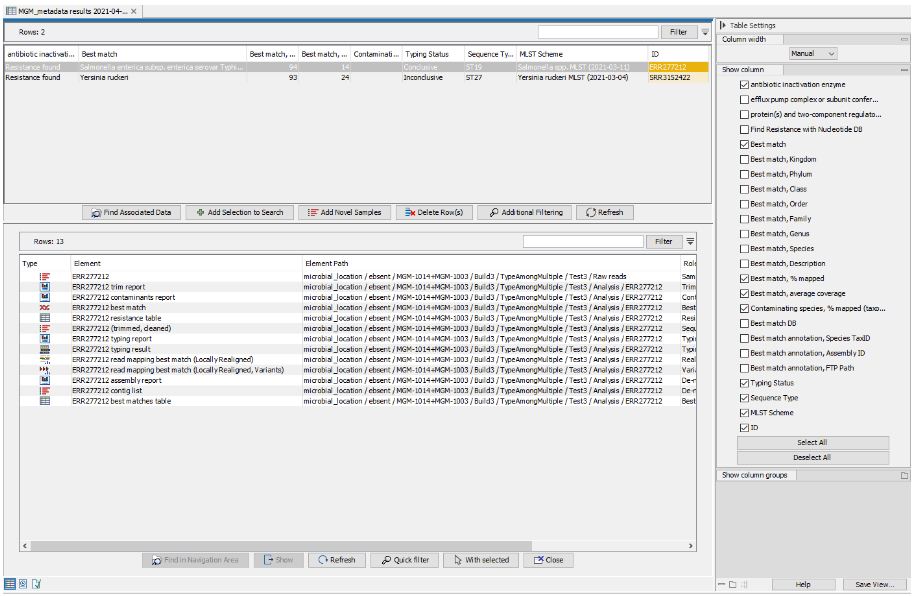
- (*)**Best match**: sequence that matches best the data according to the **Find Best Matches using K-mer Spectra** tool.
- **Matches table**: contains the best matching sequence, a list of all (maximum 100) significantly matching references and a tabular report on the various statistical values applied.
- **Read mapping best match**: output from the **Local Realignment** tool, mapping of the reads using the Best Match as reference.
- **Trimmed, cleaned sequences**: list of the sequences that were successfully trimmed and mapped to the best reference.
- **Assembly summary report**: see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=De_novo_assembly_report.html.
- **Contig list**: contig list from the **De novo assembly** tool.
- (*)**Contig list resistance table**: result table from the **Find Resistance with Nucleotide Database** tool, reports the found resistance.
- (*)**Typing report**: output from the **Type with MLST Scheme** tool, includes information on which MLST scheme was applied, the best matching sequence type (ST) as well as an overview table with sample information and a table summarizing the allele calls.
- **Typing result**: output from the **Type with MLST Scheme** tool, includes information on kmer fractions, kmer hit counts and allele count, identified and called.
- **Variant Track**: output from the **Fixed Ploidy Variant Detection** tool. Note that it is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter. This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

For each batch analysis run, the following outputs are generated:

- **Combined report**: combines the information from the trim report and MLST typing report.
- **Results metadata table**: a table containing summary information for each sample analyzed and a quick way to find the associated files. In addition, an extra column in the Result Metadata Table called "Best match, average coverage" helps the user to decide if a best match is significant, well covered and of good quality. This is especially helpful when a sample has low quality but is not contaminated.

Example of results obtained using the Type Among Multiple Species workflow

The following example includes typing of 2 samples: 1 *Salmonella enterica* (acc no. ERR277212), and 1 *Yersinia ruckeri* (acc no SRR3152422). Using the workflow **Type Among Multiple Species** workflow, analysis results are automatically summarized in the Result Metadata Table as shown in figure 2.55. The analysis results in this example include resistance found for antibiotic inactivation enzyme, name of the best matching reference, applied MLST scheme, detected sequence type and typing status. You could also choose (using options in the Table Setting



MGM_metadata results 2021-04-...

Rows: 2

antibiotic inactivati...	Best match	Best match, ...	Best match, ...	Contaminati...	Typing Status	Sequence Ty...	MLST Scheme	ID
Resistance found	Salmonella enterica subsp. enterica serovar Typhi...	94	14		Conclusive	ST19	Salmonella spp. MLST (2021-03-11)	ERR277212
Resistance found	Yersinia ruckeri	93	24		Inconclusive	ST27	Yersinia ruckeri MLST (2021-03-04)	SRR3152422

Rows: 13

Type	Element	Element Path	Role
	ERR277212	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Raw reads	Sam
	ERR277212 trim report	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Trim
	ERR277212 contaminants report	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Cont
	ERR277212 best match	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Best
	ERR277212 resistance table	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Resi
	ERR277212 (trimmed, cleaned)	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Sequ
	ERR277212 typing report	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Typi
	ERR277212 typing result	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Typi
	ERR277212 read mapping best match (Locally Realigned)	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Real
	ERR277212 read mapping best match (Locally Realigned, Variants)	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Vari
	ERR277212 assembly report	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	De-n
	ERR277212 contig list	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	De-n
	ERR277212 best matches table	microbial_location / ebsent / MGM-1014+MGM-1003 / Build3 / TypeAmongMultiple / Test3 / Analysis / ERR277212	Best

Table Settings

Column width: Manual

Show column

☑ antibiotic inactivation enzyme
☐ efflux pump complex or subunit confer...
☐ protein(s) and two-component regulato...
☐ Find Resistance with Nucleotide DB
☑ Best match
☐ Best match, Kingdom
☐ Best match, Phylum
☐ Best match, Class
☐ Best match, Order
☐ Best match, Family
☐ Best match, Genus
☐ Best match, Species
☐ Best match, Description
☑ Best match, % mapped
☑ Best match, average coverage
☑ Contaminating species, % mapped (taxo...
☐ Best match DB
☐ Best match annotation, Species TaxID
☐ Best match annotation, Assembly ID
☐ Best match annotation, FTP Path
☑ Typing Status
☑ Sequence Type
☑ MLST Scheme
☑ ID

Figure 2.55: *View of Result Metadata Table once the Type Among Multiple Species workflow has been executed (top) and associated data elements found (bottom).*

window) to display additional information such as the 'Best Match, Species' and 'Find Resistance With Nucleotide Database' for example.

Analyzing samples in batch will produce a large amount of output files, making it necessary to filter for the information you are looking for. Through the Result Metadata Table, it is possible to filter among sample metadata and analysis results. By clicking **Find Associated Data** (🔍) and optionally performing additional filtering, it is possible to perform additional analyses on a selected subset directly from this Table, such as:

- Generation of SNP trees based on the same reference used for read mapping and variant detection (section 9.1).
- Generation of K-mer Trees for identification of the closest common reference across samples (section 9.2).
- Run validated workflows (workflows that are associated with a Result Metadata Table and saved in your Navigation Area).

Note that the tool will output, among other files, variant tracks. It is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter.This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

2.3.5 Type a Known Species

The **Type a Known Species** workflow is designed for typing of samples representing a single known species (figure 2.56). It identifies the associated MLST, determines variants found when mapping the sample data against the user specified reference, and finds occurring resistance genes if they match genes within the user specified resistance database.

Preliminary steps to run the Type a Known Species workflow

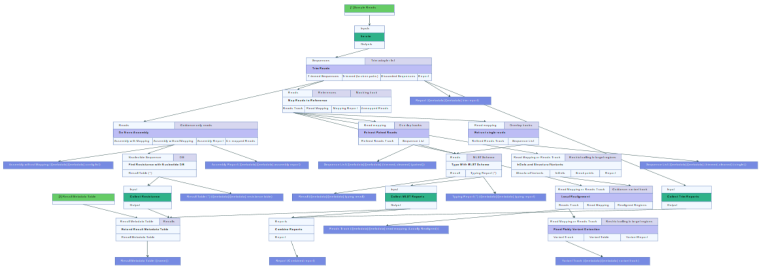
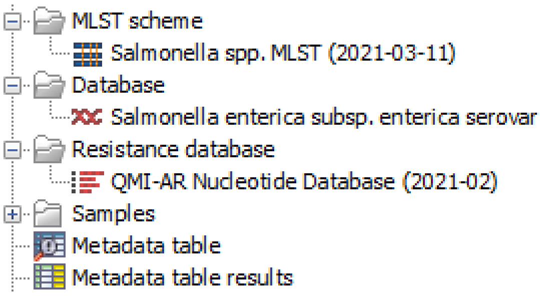


Figure 2.56: *Overview of the template Type a Known Species workflow.*

Before starting the workflow,

- Download microbial genomes using either the **Download Custom Microbial Reference Database**tool, the prokaryotic databases from the **Download Curated Microbial Reference Database** tool or the **Download Pathogen Reference Database** tool (see section VI). Databases can also be created using the **Update Sequence Attributes in Lists** tool.
- Download the MLST schemes using the **Download MLST Schemes** tool (see section 13.2).
- Download the database for the **Find Resistance with Nucleotide Database** tool using the **Download Resistance Database** tool (see section 17.1).
- Create a New Result Metadata table using the **Create Result Metadata Table** tool (see section 19.2.1).

When you are ready to start the workflows, your navigation area should look similar to the figure 2.57 with one MLST scheme and one reference genome.



MLST scheme
Salmonella spp. MLST (2021-03-11)
Database
Salmonella enterica subsp. enterica serovar
Resistance database
QMI-AR Nucleotide Database (2021-02)
Samples
Metadata table
Metadata table results

Figure 2.57: *Overview of the Navigation area after creating the result metadata table and downloading the databases and MLST scheme necessary to run the workflow.*

How to run the Type a Known Species workflow

To run the workflow, go to:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Typing and Epidemiology** () | **Type a Known Species** ()

1. Specify the **sample(s)** or folder(s) of samples you would like to type (figure 2.58) and click **Next**. Remember that if you select several items, they will be run as batch units.
2. Specify the **Result Metadata Table** you want to add your results to (figure 2.59) and click **Next**.

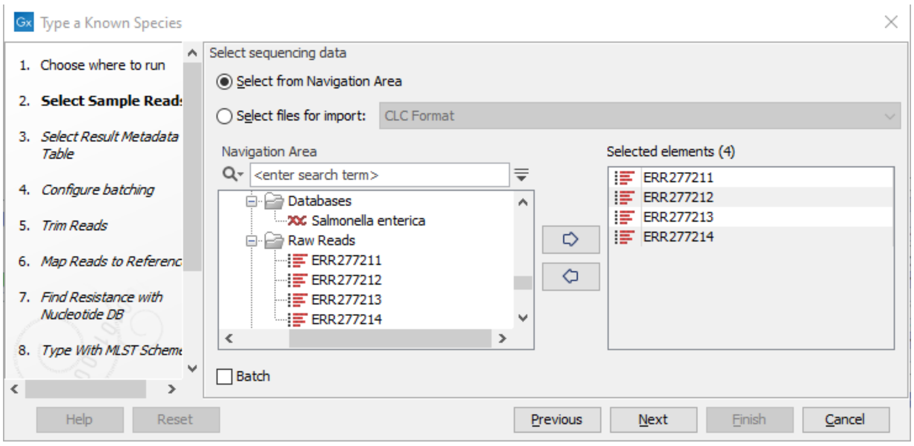


Figure 2.58: *Select the reads from the sample(s) you would like to type.*

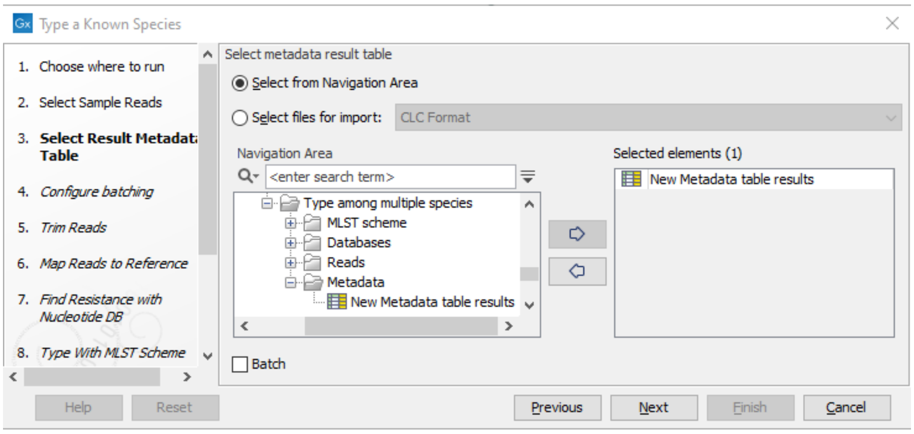

Type a Known Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadat: Table
4. Configure batching
5. Trim Reads
6. Map Reads to Reference
7. Find Resistance with Nucleotide DB
8. Type With MLST Scheme
Select metadata result table
● Select from Navigation Area
○ Select files for import: CLC Format
Navigation Area
<enter search term>
Type among multiple species
MLST scheme
Databases
Reads
Metadata
New Metadata table results
Selected elements (1)
New Metadata table results
□ Batch
Help
Reset
Previous
Next
Finish
Cancel

Figure 2.59: *Select the metadata table you would like to use.*

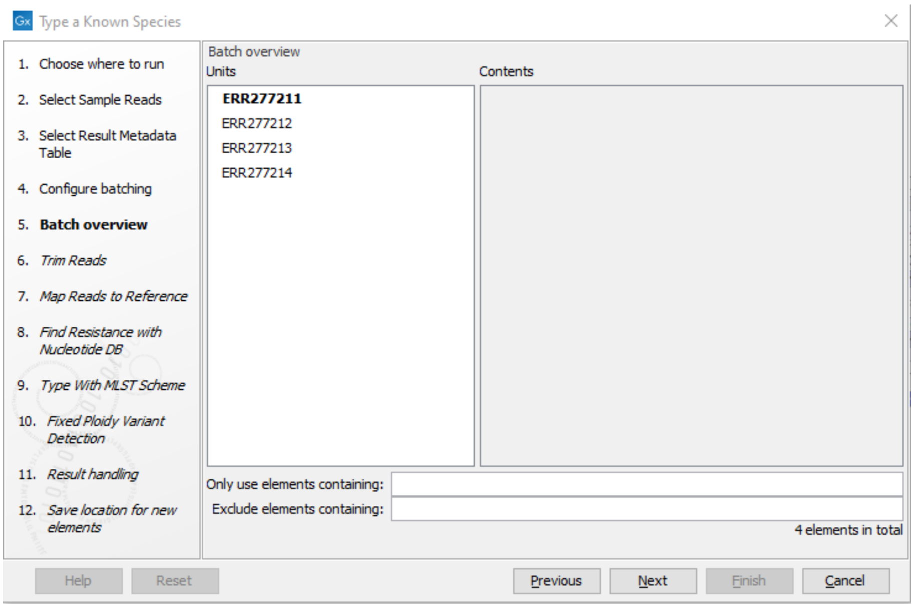
3. Define batch units using organisation of input data to create one run per input or use a metadata table to define batch units. Click **Next**.

4. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 2.60).

5. You can specify a **trim adapter list** and set up parameters if you would like to trim your sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 2.61). Learn about trim adapter lists at `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_adapter_list.html`.

   The parameters that can be set are:

   - **Trim ambiguous nucleotides**: if checked, this option trims the sequence ends based on the presence of ambiguous nucleotides (typically N).
   - **Maximum number of ambiguities**: defines the maximal number of ambiguous nucleotides allowed in the sequence after trimming.
   - **Trim using quality scores**: if checked, and if the sequence files contain quality scores from a base-caller algorithm, this information can be used for trimming sequence ends.



Batch overview

Units: ERR277211, ERR277212, ERR277213, ERR277214

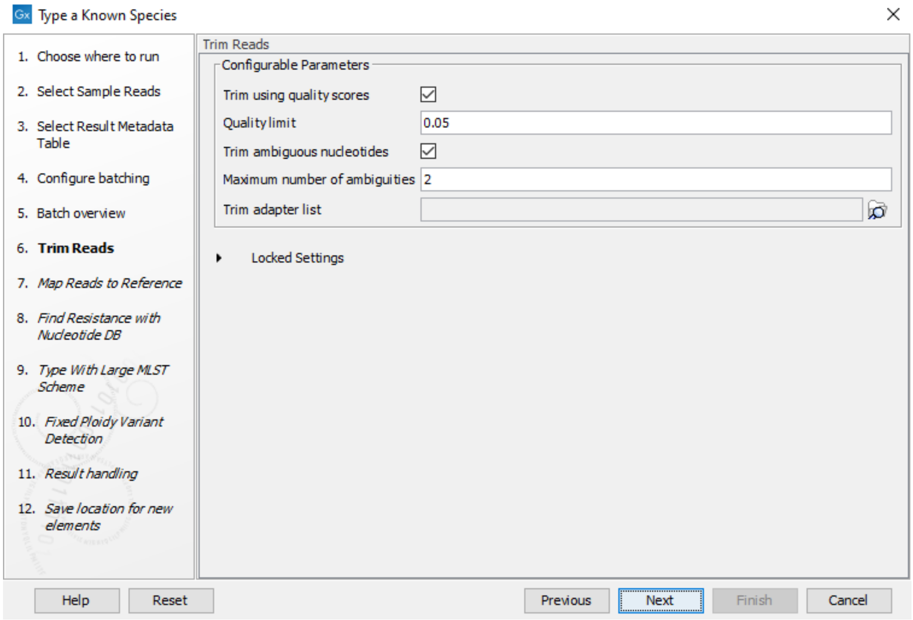
Contents

Only use elements containing:

Exclude elements containing:

4 elements in total

Figure 2.60: *Choose which of the samples present in the selected folder(s) you want to analyze.*


Type a Known Species
1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Map Reads to Reference
8. Find Resistance with Nucleotide DB
9. Type With Large MLST Scheme
10. Fixed Ploidy Variant Detection
11. Result handling
12. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Trim Reads

Configurable Parameters

Trim using quality scores ☑

Quality limit 0.05

Trim ambiguous nucleotides ☑

Maximum number of ambiguities 2

Trim adapter list

Locked Settings

Figure 2.61: *You can choose to trim adapter sequences from your sequencing reads.*

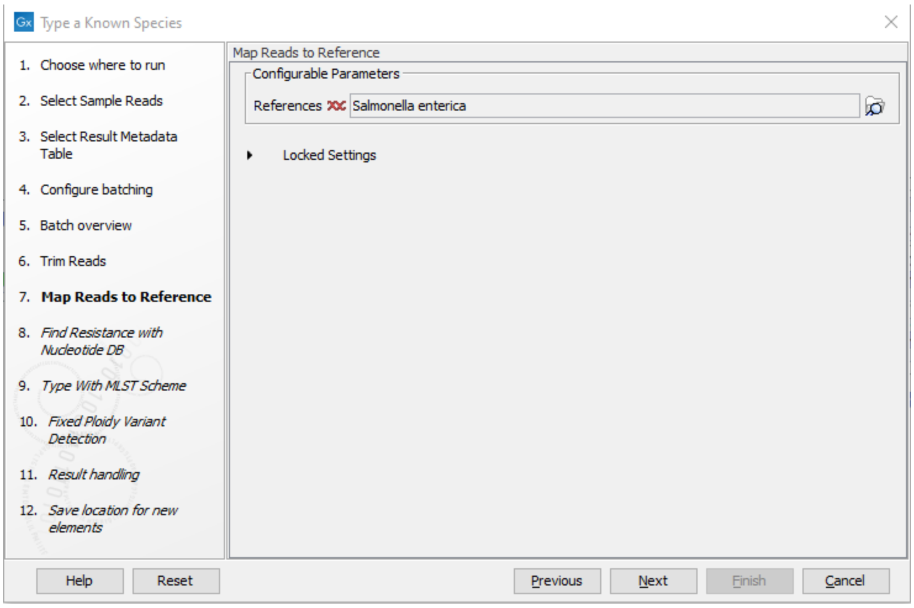
- **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.

Click **Next**.

6. Choose the **species reference** to be used by the **Map reads to reference** tool (figure 2.62). Click **Next**.

7. Specify the **resistance database** (figure 2.63) and set the parameters for the **Find Resistance with Nucleotide Database** tool.

   The parameters that can be set are:



Type a Known Species

1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. **Map Reads to Reference**
8. *Find Resistance with Nucleotide DB*
9. *Type With MLST Scheme*
10. *Fixed Ploidy Variant Detection*
11. *Result handling*
12. *Save location for new elements*

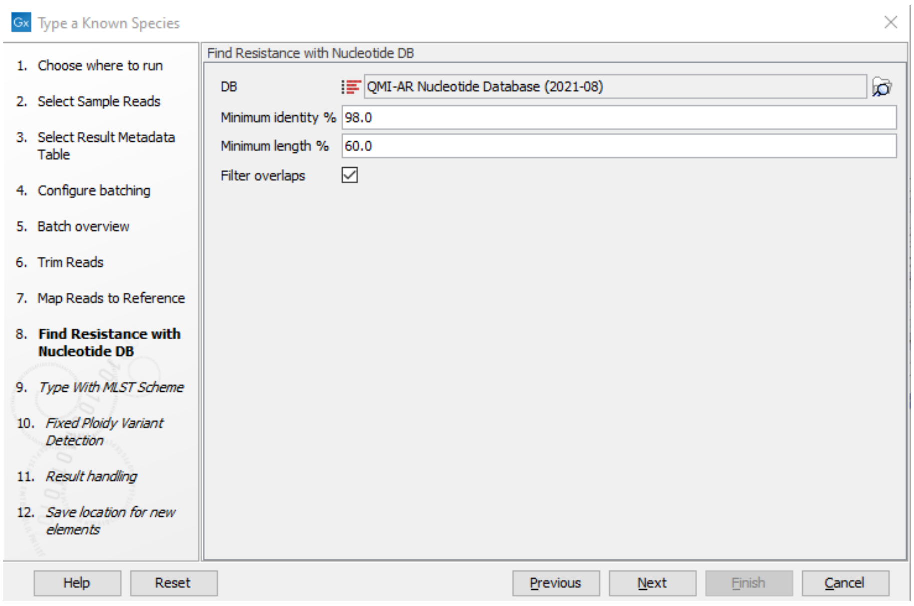
Map Reads to Reference

Configurable Parameters

References: Salmonella enterica

Locked Settings

Figure 2.62: *Specify the reference for the Map reads to reference tool.*



Type a Known Species

1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Map Reads to Reference
8. **Find Resistance with Nucleotide DB**
9. *Type With MLST Scheme*
10. *Fixed Ploidy Variant Detection*
11. *Result handling*
12. *Save location for new elements*

Find Resistance with Nucleotide DB

DB: QMI-AR Nucleotide Database (2021-08)

Minimum identity %: 98.0

Minimum length %: 60.0

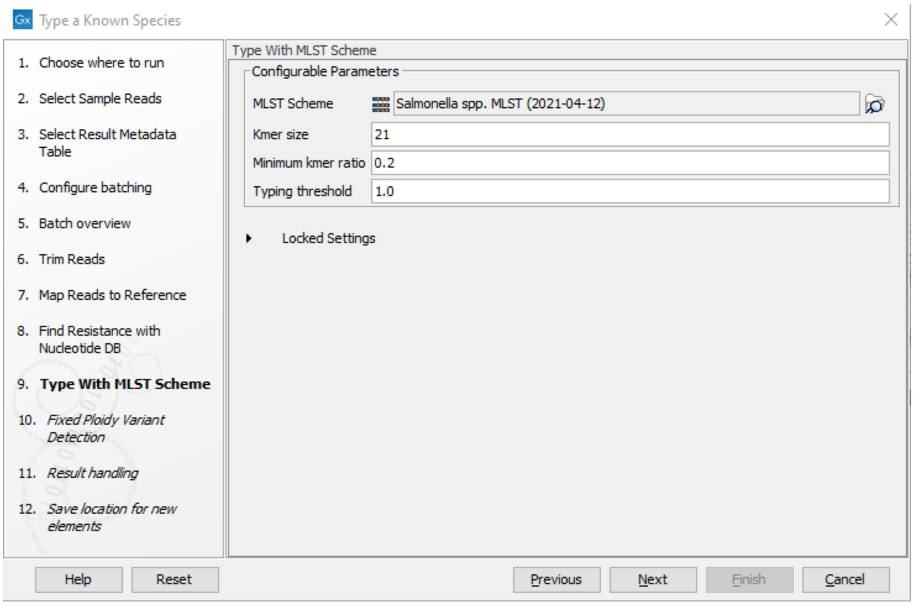
Filter overlaps: ☑

Figure 2.63: *Specify the resistance database to be used for the Find Resistance with Nucleotide Database tool.*

- **Minimum Identity** %: is the threshold for the minimum percentage of nucleotides that are identical between the best matching resistance gene in the database and the corresponding sequence in the genome.
- **Minimum Length** %: reflect the percentage of the total resistance gene length that a sequence must overlap a resistance gene to count as a hit for that gene. Here represented as a percentage of the total resistance gene length.
- **Filter overlaps**: will perform extra filtering of results per contig, where one hit is contained by the other with a preference for the hit with the higher number of aligned nucleotides (length * identity).

Click **Next**.

8. Specify the parameters for the **Type with MLST Scheme** tool (figure 2.64).



MLST Scheme: Salmonella spp. MLST (2021-04-12)
Kmer size: 21
Minimum kmer ratio: 0.2
Typing threshold: 1.0

Figure 2.64: *Specify the parameters for MLST typing.*

The parameters that can be set are:

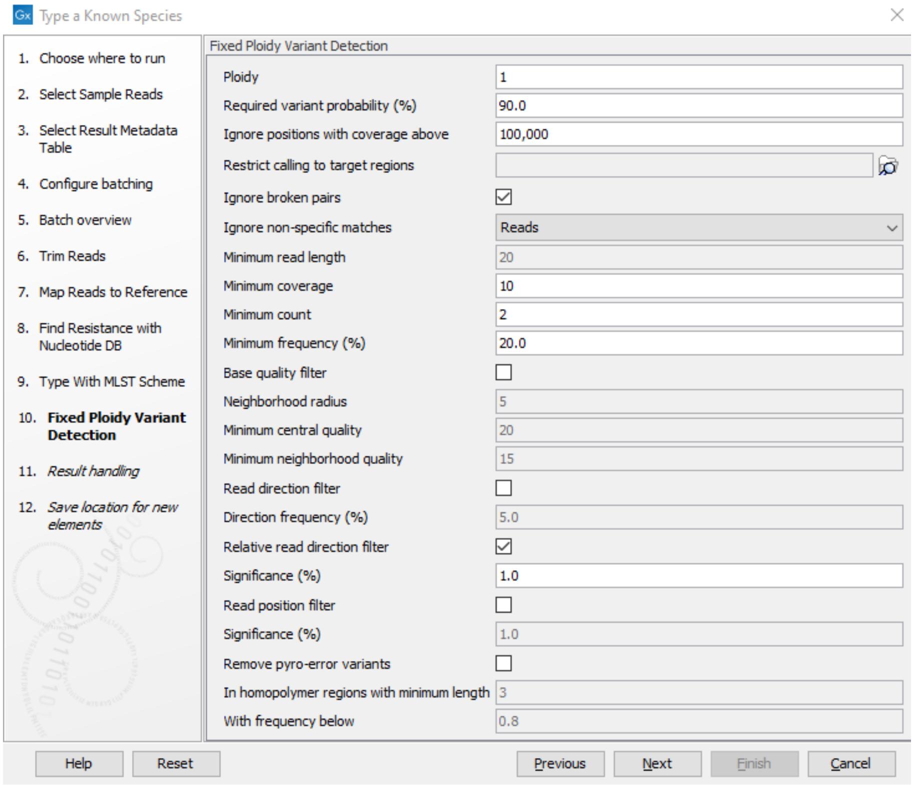
- **Kmer size**: determines the number of nucleotides in the kmer - raising this setting might increase specificity at the cost of some sensitivity.
- **Typing threshold**: determines how many of the kmers in a sequence type that needs to be identified before a typing is considered conclusive. The default setting of 1.0 means that all kmers in all alleles must be matched.
- **Minimum kmer ratio**: the minimum kmer ratio of the least occurring kmer and the average kmer hit count. If an allele scores higher than this threshold it is classified as a high-confidence call.
- **Typing threshold**: The typing threshold determines how many of the kmers in a sequences type that needs be identified before a typing is considered conclusive. The default setting of 1.0 means that all kmers in all alleles must be matched. Lowering the setting of 0.99 would mean that on avergae 99% of all kmers in all the alleles of a given sequence type must be detected before the sequence type is considered conclusive.

Click **Next**.

9. Specify the parameters for the **Fixed Ploidy Variant Detection** tool (figure 2.65) before clicking **Next**.

The parameters that can be set are:

- **Ploidy**: The expected ploidy of the species of interest.
- **Required variant probability (%)**: 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



1. Choose where to run
2. Select Sample Reads
3. Select Result Metadata Table
4. Configure batching
5. Batch overview
6. Trim Reads
7. Map Reads to Reference
8. Find Resistance with Nucleotide DB
9. Type With MLST Scheme
10. **Fixed Ploidy Variant Detection**
11. *Result handling*
12. *Save location for new elements*

Fixed Ploidy Variant Detection

Ploidy	1
Required variant probability (%)	90.0
Ignore positions with coverage above	100,000
Restrict calling to target regions	
Ignore broken pairs	☑
Ignore non-specific matches	Reads
Minimum read length	20
Minimum coverage	10
Minimum count	2
Minimum frequency (%)	20.0
Base quality filter	□
Neighborhood radius	5
Minimum central quality	20
Minimum neighborhood quality	15
Read direction filter	□
Direction frequency (%)	5.0
Relative read direction filter	☑
Significance (%)	1.0
Read position filter	□
Significance (%)	1.0
Remove pyro-error variants	□
In homopolymer regions with minimum length	3
With frequency below	0.8

Figure 2.65: *Specify the parameters to be used for the Fixed Ploidy Variant Detection tool.*

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 positions with coverage above**: Ignore positions with a read-coverage larger than this value.
- **Restrict calling to target regions**: Select a region track to specify the regions in which variants should be called.
- **Ignore broken pairs**: You can choose to ignore broken pairs by clicking this option.
- **Ignore non-specific matches**: You can choose to ignore non-specific matches between reads, regions or to not ignore them at all.
- **Minimum read length**: Only variants in reads longer than this size are called.
- **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.
- **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.
- **Neighborhood radius**: Determine how far away from the current variant the quality assessment should extend.

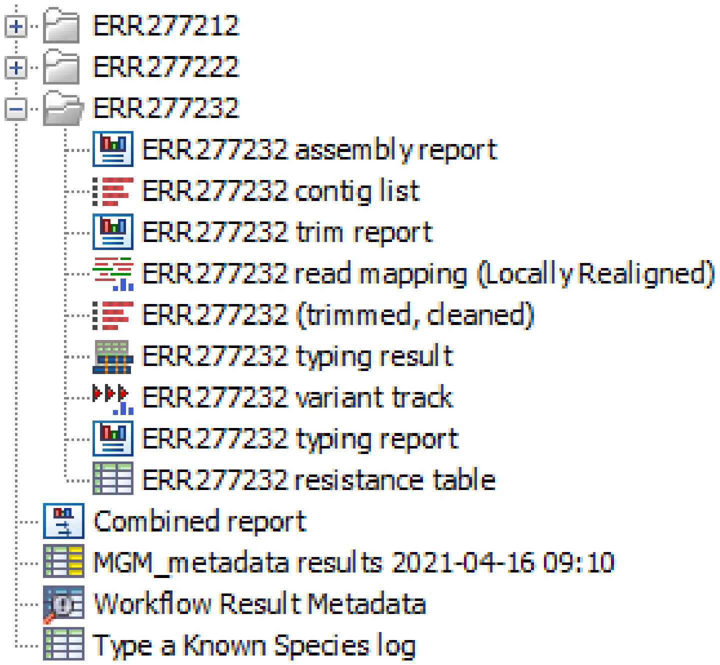
- **Minimum central quality**: Reads whose central base has a quality below the specified value will be ignored. This parameter does not apply to deletions since there is no "central base" in these cases.
- **Minimum neighborhood quality**: Reads for which the minimum quality of the bases is below the specified value will be ignored.
- **Read direction filters**: The read direction filter removes variants that are almost exclusively present in either forward or reverse reads.
- **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.
- **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**: This filter 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.
- **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.
- **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 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.

Click **Next**.

10. In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters. Choose to save the results (we recommend to create a new folder for it) and click **Finish**.

The output will be saved in the new folder you created (figure 2.66), but those marked with a (*) in the list below will also be added automatically to the Metadata Result table.

- **Trim report**: report from the **Trim Sequences** tool (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html).
- **Read mapping**: output from the **Local Realignment** tool, mapping of the reads to the specified reference.



ERR277212
ERR277222
ERR277232
ERR277232 assembly report
ERR277232 contig list
ERR277232 trim report
ERR277232 read mapping (Locally Realigned)
ERR277232 (trimmed, cleaned)
ERR277232 typing result
ERR277232 variant track
ERR277232 typing report
ERR277232 resistance table
Combined report
MGM_metadata results 2021-04-16 09:10
Workflow Result Metadata
Type a Known Species log

Figure 2.66: *Output files from the Type a Known Species workflow.*

- **Trimmed, cleaned sequences**: list of the sequences that were successfully trimmed and mapped to the best reference.

- **Assembly summary report**: see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=De_novo_assembly_report.html.

- **Contig list**: contig list from the **De novo assembly** tool.

- (*)**Contig list resistance table**: result table from the **Find Resistance with Nucleotide Database** tool, reports the found resistance.

- (*)**Typing report**: output from the **Type with MLST Scheme** tool, includes information on the applied MLST scheme, the best matching sequence type (ST) as well as an overview table with sample information and a table summarizing the allele calls.

- **Typing result**: output from the **Type with MLST Scheme** tool, includes information on kmer fractions, kmer hit counts and allele count, identified and called.

- **Variant Track**: output from the **Fixed Ploidy Variant Detection** tool. Note that it is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter.This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

For each batch analysis run, the following outputs are generated:

- **Combined report**: combines the information from the trim report and MLST typing report.

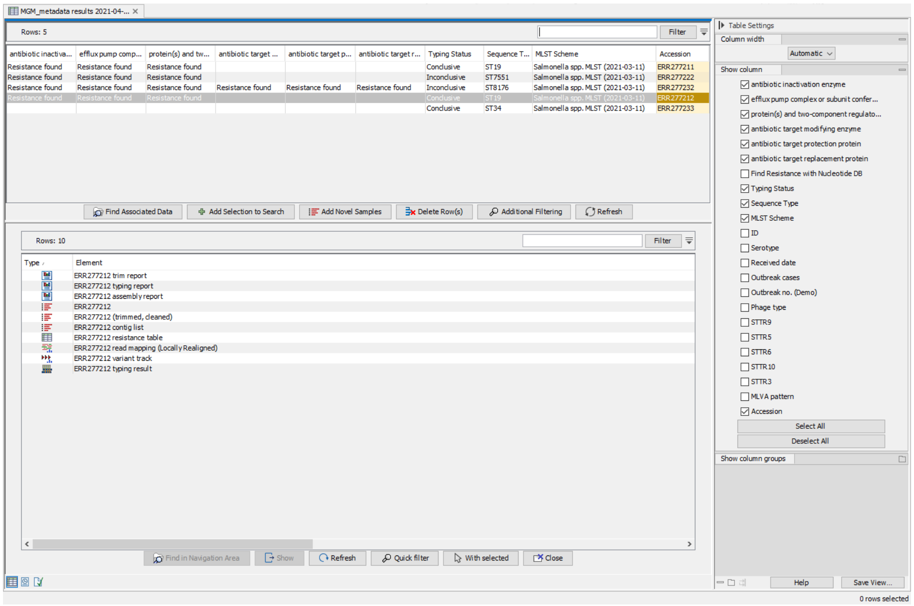
- **Results metadata table**: An table containing summary information for each sample analyzed and a quick way to find the associated files. In addition, an extra column in the Result

Metadata Table called "Best match, average coverage" helps the user to decide if a best match is significant, well covered and of good quality. This is especially helpful when a sample has low quality but is not contaminated.

Example of results obtained using the Type a Known Species workflow

In this section, sequence data from five *Salmonella enterica* samples were typed using a customized *Salmonella enterica* version of the Type a Known Species workflow: the specified reference and the MLST scheme was specified as *Salmonella enterica*. The created Result Metadata Table shows these five samples and associated metadata for sample with accession no ERR277212.

Once the workflow analysis was performed, additional columns listing the analysis results (e.g., MLST, best matching reference and identified resistance genes) have automatically been added to the Result Metadata Table (figure 2.67). To ease the overview regarding applied analysis parameters, columns stating the workflow specified reference list, MLST scheme and resistance gene database have also been added automatically to the table.



Rows: 5

antibiotic inactiva...	efflux pump comp...	protein(s) and tw...	antibiotic target ...	antibiotic target p...	antibiotic target r...	Typing Status	Sequence T...	MLST Scheme	Accession
Resistance found	Resistance found	Resistance found				Conclusive	ST19	Salmonella spp. MLST (2021-03-11)	ERR277211
Resistance found	Resistance found	Resistance found				Inconclusive	ST7551	Salmonella spp. MLST (2021-03-11)	ERR277222
Resistance found	Resistance found	Resistance found	Resistance found	Resistance found	Resistance found	Inconclusive	ST8176	Salmonella spp. MLST (2021-03-11)	ERR277232
Resistance found	Resistance found	Resistance found				Conclusive	ST19	Salmonella spp. MLST (2021-03-11)	ERR277212
						Conclusive	ST34	Salmonella spp. MLST (2021-03-11)	ERR277233

Rows: 10

Type	Element
	ERR277212 trim report
	ERR277212 typing report
	ERR277212 assembly report
	ERR277212
	ERR277212 (trimmed, cleaned)
	ERR277212 contig list
	ERR277212 resistance table
	ERR277212 read mapping (Locally Realigned)
	ERR277212 variant track
	ERR277212 typing result

Column width: Automatic

Show column
☑ antibiotic inactivation enzyme
☑ efflux pump complex or subunit confer...
☑ protein(s) and two-component regulato...
☑ antibiotic target modifying enzyme
☑ antibiotic target protection protein
☑ antibiotic target replacement protein
☐ Find Resistance with Nucleotide DB
☑ Typing Status
☑ Sequence Type
☑ MLST Scheme
☐ ID
☐ Serotype
☐ Received date
☐ Outbreak cases
☐ Outbreak no. (Demo)
☐ Phage type
☐ STTR9
☐ STTR5
☐ STTR6
☐ STTR10
☐ STTR3
☐ MLVA pattern
☑ Accession
Select All
Deselect All

Show column groups

0 rows selected

Figure 2.67: *View of the updated Result Metadata Table and associated elements once the Type a Known Species workflow has been executed on the* Salmonella *(acc no ERR277212) sample.*

Analyzing samples in batch will produce a large amount of output files, making it necessary to filter for the information you are looking for. Through the Result Metadata Table, it is possible to filter among sample metadata and analysis results. By clicking **Find Associated Data** ( ) and optionally performing additional filtering, it is possible to perform additional analyses on a selected subset directly from this Table, such as:

- Generation of SNP trees based on the same reference used for read mapping and variant detection (section 9.1).
- Generation of K-mer Trees for identification of the closest common reference across samples (section 9.2).
- Run validated workflows (workflows that are associated with a Result Metadata Table and saved in your Navigation Area).

For more information on filtering and running analysis directly from a Result Metadata Table section, see section 19.2.2.

Note that the tool will output, among other files, variant tracks. It is possible to export multiple variant track files from monoploid data into a single VCF file with the Multi-VCF exporter. This exporter becomes available when installing the CLC Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

2.4 QIAseq Analysis template workflows

The QIAseq Analysis template workflows are found at:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **QIAseq Analysis** ( )

The QIAseq template workflows are configured with workflow roles and therefore require reference data sets containing elements with the relevant roles.
To learn about workflow roles, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Configuring_input_output_elements.html.

QIAGEN reference data sets are available from **QIAGEN Sets Reference Data Library** accessible via **References ( )** in the top Toolbar. Alternatively, access it from the **Utilities** menu and select **Manage Reference Data ( )**.
To learn about QIAGEN reference sets, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QIAGEN_Sets.html.

Launching the QIAseq Analysis template workflows

The following sections describe how to launch the QIAseq template workflows from the **Toolbox**. The workflows are also available from the **QIAseq Panel Analysis Assistant** in the *xHYB Viral and Bacterial* category (see section 2.5).

2.4.1 Analyze QIAseq xHYB Viral Panel Data (Human host)

The Analyze QIAseq xHYB Viral Panel Data (Human host) template workflow trims reads, performs taxonomic profiling, and calls viral variants. It is suitable for analysis of human data generated with the QIAseq xHYB viral panels:

- QIAseq xHYB Respiratory Panel
- QIAseq xHYB Viral STI Panel
- QIAseq xHYB Adventitious Agent Panel
- QIAseq xHYB MPXV Panel

To analyze non-human samples, you can create a copy of the workflow and edit it to fit your specific application, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Template_workflows.html. Since the workflow element **Map Reads to Human Control Genes** is relevant for human data only, you should delete this. In addition, if a host genome is not relevant for you application, open the **Taxonomic Profiling** workflow element, and uncheck *Filter host reads*.

Once the workflow copy is customized, you can install it to make it available from the **Toolbox**, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Installing_workflow.html.

QIAGEN reference data set

The *QIAseq xHYB Viral Panels* reference data set is available from **QIAGEN Sets Reference Data Library** accessible via **References ()** in the top Toolbar.

Like the template workflow, the reference data set is designed for human samples. It contains both a human host taxonomic profiling index, and a sequence list with human control genes for use in the workflow step **Map Reads to Human Control Genes**.

For analysis of non-human data, if a host is relevant for your application, you can create a host taxonomic profiling index from your host reference genome using **Create Taxonomic Profiling Index**, see section 15.4.

Launching the workflow

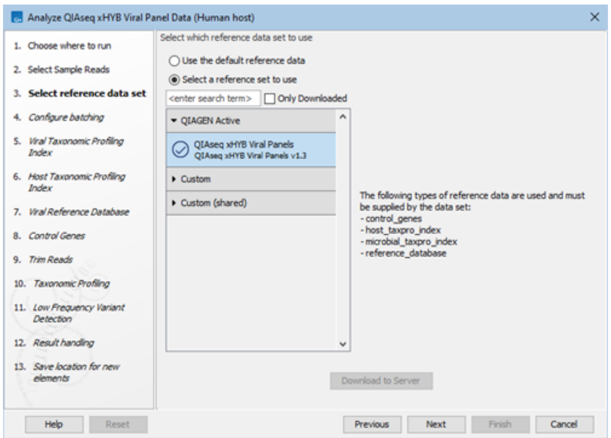
The Analyze QIAseq xHYB Viral Panel Data (Human host) workflow is at:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **QIAseq Analysis** () | **Analyze QIAseq xHYB Viral Panel Data (Human host)** ()

Launch the workflow and step through the wizard.

1. Select the sequence list(s) containing the reads to analyze. Click on **Next**.
2. Select a reference data set or select "Use the default reference data" to configure the reference data elements individually in subsequent wizard steps (figure 2.68). Click on **Next**.
3. Choose whether batch units should be defined based on organization of the input data, or by provided metadata (figure 2.69). For information on how to use metadata when running part of a workflow multiple times, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Running_part_workflow_multiple_times.html.
4. Next, you can review the batch units resulting from your selections above. Click on **Next**.
5. Verify or select the viral taxonomic profiling index (figure 2.70) and click on **Next**.
6. Verify or select the host taxonomic profiling index and click on **Next**.
7. Select the viral reference database(s). If in the first step you selected the QIAseq xHYB Viral Panels reference set, you can now select which of the available viral reference databases from that set to apply (2.71). If you chose to use the default reference data, select a reference database and click on **Next**.
8. Verify or select the control genes and click on **Next**.
9. Specify the trim settings.
10. Specify Taxonomic Profiling settings (figure 2.72).
11. Specify Low Frequency Variant Detection settings, see figure 2.73.

12. Finally, select a location to save outputs to and click on **Finish**.


Analyze QIAseq xHYB Viral Panel Data (Human host)
1. Choose where to run
2. Select Sample Reads
3. Select reference data set
4. Configure batching
5. Viral Taxonomic Profiling Index
6. Host Taxonomic Profiling Index
7. Viral Reference Database
8. Control Genes
9. Trim Reads
10. Taxonomic Profiling
11. Low Frequency Variant Detection
12. Result handling
13. Save location for new elements
QIAGEN Active
Custom
Custom (shared)
Download to Server
Help
Reset
Previous
Next
Finish
Cancel

Select which reference data set to use

○ Use the default reference data

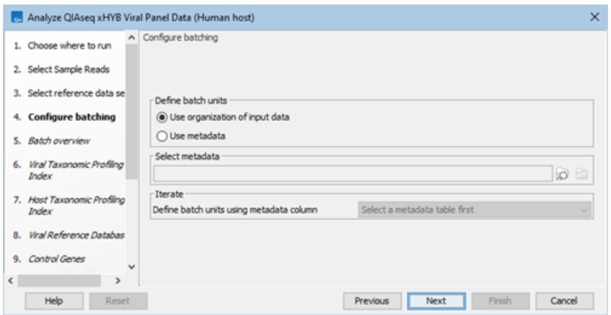
● Select a reference set to use

<enter search term> □ Only Downloaded

QIAseq xHYB Viral Panels
QIAseq xHYB Viral Panels v1.3

The following types of reference data are used and must be supplied by the data set:
- control_genes
- host_taxpro_index
- microbial_taxpro_index
- reference_database

Figure 2.68: *Select reference data set.*


Analyze QIAseq xHYB Viral Panel Data (Human host)
1. Choose where to run
2. Select Sample Reads
3. Select reference data se
4. Configure batching
5. Batch overview
6. Viral Taxonomic Profiling Index
7. Host Taxonomic Profiling Index
8. Viral Reference Databas
9. Control Genes
Help
Reset
Previous
Next
Finish
Cancel

Configure batching

Define batch units

● Use organization of input data

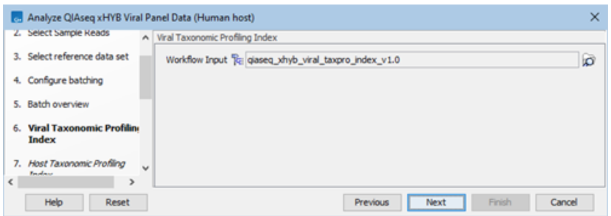
○ Use metadata

Select metadata

Iterate

Define batch units using metadata column: Select a metadata table first

Figure 2.69: *Define batch units.*


Analyze QIAseq xHYB Viral Panel Data (Human host)
2. Select Sample Reads
3. Select reference data set
4. Configure batching
5. Batch overview
6. Viral Taxonomic Profiling Index
7. Host Taxonomic Profiling Index
Help
Reset
Previous
Next
Finish
Cancel

Viral Taxonomic Profiling Index

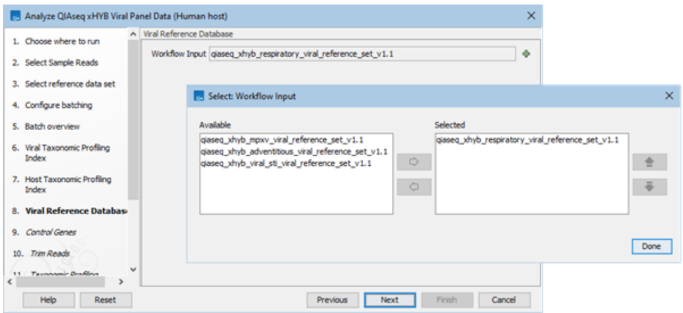
Workflow Input: qiaseq_xhyb_viral_taxpro_index_v1.0

Figure 2.70: *Select viral taxonomic profiling index.*

Workflow tools and outputs

The Analyze QIAseq xHYB Viral Panel Data (Human host) template workflow consists of the following tools.

- **QC for Sequencing Reads.** Performs basic quality control of the sequencing reads. The output, which is included in a combined report, can be used to evaluate the quality of the sequencing reads. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html.
- **Trim Reads.** Removes adapter sequences and low quality nucleotides. The appropriate settings for the Trim Reads tool depends on the protocol used to generate



Analyze QIAseq xHYB Viral Panel Data (Human host)

Viral Reference Database

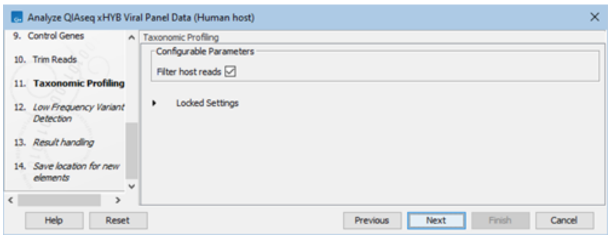
Workflow Input: qiaseq_xhyb_respiratory_viral_reference_set_v1.1

Select: Workflow Input

Available:
qiaseq_xhyb_mpxv_viral_reference_set_v1.1
qiaseq_xhyb_adventitious_viral_reference_set_v1.1
qiaseq_xhyb_viral_sti_viral_reference_set_v1.1

Selected:
qiaseq_xhyb_respiratory_viral_reference_set_v1.1

Figure 2.71: *Select one or more viral reference databases.*



Analyze QIAseq xHYB Viral Panel Data (Human host)

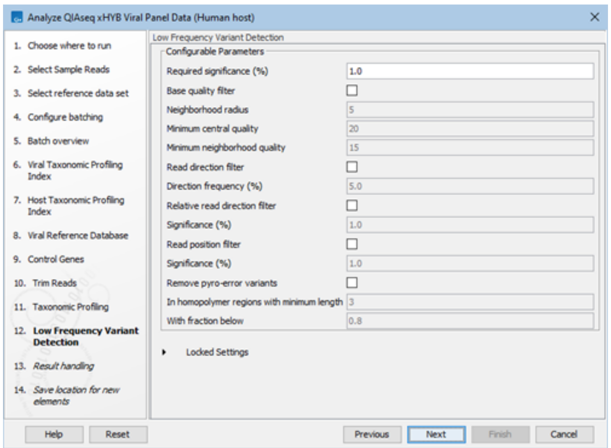
Taxonomic Profiling

Configurable Parameters

Filter host reads ☑

Locked Settings

Figure 2.72: *Set taxonomic profiling parameters.*



Analyze QIAseq xHYB Viral Panel Data (Human host)

Low Frequency Variant Detection

Configurable Parameters

Required significance (%)	1.0
Base quality filter	☐
Neighborhood radius	5
Minimum central quality	20
Minimum neighborhood quality	15
Read direction filter	☐
Direction frequency (%)	5.0
Relative read direction filter	☐
Significance (%)	1.0
Read position filter	☐
Significance (%)	1.0
Remove pyro-error variants	☐
In homopolymer regions with minimum length	3
With fraction below	0.8

Locked Settings

1. Choose where to run
2. Select Sample Reads
3. Select reference data set
4. Configure batching
5. Batch overview
6. Viral Taxonomic Profiling Index
7. Host Taxonomic Profiling Index
8. Viral Reference Database
9. Control Genes
10. Trim Reads
11. Taxonomic Profiling
12. Low Frequency Variant Detection
13. Result handling
14. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Figure 2.73: *Low frequency variant detection settings.*

the reads. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_Reads.html.

- **Taxonomic Profiling**. Analyzes the taxonomic composition of samples and estimates the relative abundance of the detected taxa. See section 5.2. Host reads i.e., reads that map to the host taxonomic profiling index, do not count toward the taxonomic profiling result, but are used as input for **Map Reads to Human Control Genes**. Viral reads - reads that map to the viral taxonomic profiling index - are later used as input for **Find Best References using Read Mapping**.

- **Map Reads to Human Control Genes**. Maps the host reads output from **Taxonomic Profiling** to the host taxonomic profiling index, to a reference of human control genes. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Map_Reads_Reference.html. This serves as a QC step to verify mapping to the human control genes. For human samples, you expect to see mapping of reads to all human control genes.

- **Find Best References using Read Mapping**. Maps the viral reads output from **Taxonomic Profiling** to the selected viral reference database to identify which reference sequence is the "Best match". See section 8.2.

- **Remove Duplicate Mapped Reads**. Removes duplicate reads derived from PCR amplification (or other enrichment) during sample preparation from the mapping. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Remove_Duplicate_Mapped_Reads.html. The output reads track is used as input for **Local Realignment**.

- **Local Realignment**. Improves the alignment of the reads in the reads track. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Local_Realignment.html.

- **Low Frequency Variant Detection**. Calls variants in the read mapping that are present at low frequencies. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Low_Frequency_Variant_Detection.html.

- **Filter on Custom Criteria**, **Filter against Known Variants**, and **Remove Marginal Variants**. Remove variants that fall below a set of thresholds. For this workflow, coverage >30 and frequency >20% is required. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Variant_filtering.html.

- **Amino Acid Changes**. Uses the called variants to generate a track of amino acid changes. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Amino_Acid_Changes.html.

- **Create Mapping Graph** and **Identify Graph Threshold Areas**. Creates a track with regions with coverage below a threshold. For this workflow, the threshold is set to 30. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Create_Mapping_Graph.html and https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Identify_Graph_Threshold_Areas.html.

- **Extract Consensus Sequence**. Makes a consensus sequence from the read tracks from **Local Realignment**. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Extract_Consensus_Sequence.html.

- **QC for Read Mapping**. Performs quality control of the read mapping. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Read_Mapping.html.

- **Merge Abundance Tables**. Merges the sample-specific abundance tables to one combined abundance table. See section 6.1.

The sample-specific outputs provided by this workflow are:

- **QC Report Raw Reads**. Quality control metrics for the raw reads.

- **Abundance Table**. The abundance of each identified taxa, along with their full taxonomy. See section 5.2.3.

- **Read Mapping Human Control Genes**. The host reads mapped against the control gene reference.
- **Viral Reads**. List of reads that mapped to the viral taxonomic profiling index.
- **Find Best Reference Report**. Report of the "Best match" reference identified by **Find Best References using Read Mapping**.
- **Best Match Sequence**. The "Best match" reference sequence as identified by the Find Best References using Read Mapping tool.
- **Read Mapping**. Reads mapped to the "Best match" viral reference. Output from **Local Realignment**.
- **Consensus Sequence**. Viral consensus sequence(s), extracted from the above *Read Mapping* output.
- **Annotated Variant Track**. List of detected variants left after filtering, annotated with amino acid changes.
- **Amino Acid Track**. List of amino acid changes.
- **Low Coverage Areas**. List of low coverage regions in the *Read Mapping* output.
- **Track List**. Collection of the following viral tracks: Consensus sequence, reads, variants, amino acid changes, and low coverage regions.
- **QC and Taxonomic Profiling Report** combines QC Report Raw Reads and the Taxonomic Profiling report.

The combined outputs provided by this workflow are:

- **Taxonomic Profiling Report**. Combines taxonomic profiling report content across samples in the workflow run. See section 5.2.2.
- **Human Control Genes Read Mapping Report**. Holds information about the mapping of host reads to the human control genes for all samples in the workflow run.
- **Combined Report**. Combines information from various tools, including QC, taxonomic profiling and mapping reports.
- **Merged Abundance Table**. Provides abundances for the detected taxa for all samples in the workflow run. See section 5.2.3.

2.4.2 Find QIAseq xHYB AMR Markers (Human host)

The Find QIAseq xHYB AMR Markers (Human host) template workflow trims reads and detects antimicrobial resistance (AMR) markers.

It is suitable for analysis of human data generated with the QIAseq xHYB AMR Panel.

To analyze non-human samples, you can create a copy of the workflow and edit it to fit your specific application, see `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/`

index.php?manual=Template_workflows.html. Since the workflow element **Map Reads to Human Control Genes** is relevant for human data only, you should delete this.
Once the workflow copy is customized, you can install it to make it available from the **Toolbox**, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Installing_workflow.html.

QIAGEN reference data set

The *QIAseq xHYB AMR Panel* reference data set is available from **QIAGEN Sets Reference Data Library** accessible via **References ( )** in the top Toolbar.

Launching the workflow

The Find QIAseq xHYB AMR Markers (Human host) workflow is at:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **QIAseq Analysis** ( ) | **Find QIAseq xHYB AMR Markers (Human host)** ( )

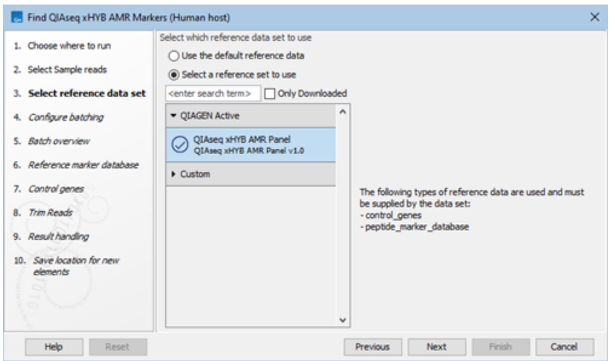
Launch the workflow and step through the wizard.

1. Select the sequence list(s) containing the reads to analyze and click on **Next**.
2. Select a reference data set or select "Use the default reference data" to configure the reference data elements individually in subsequent wizard steps (figure 2.74). Click on **Next**.
3. Choose whether batch units should be defined based on organization of the input data, or by provided metadata (figure 2.75). For information on how to use metadata when running part of a workflow multiple times, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Running_part_workflow_multiple_times.html.
4. Next, you can review the batch units resulting from your selections above. Click on **Next**.
5. Verify or select the reference marker database (figure 2.76). Click on **Next**.
6. Verify or select control genes (figure 2.77). Click on **Next**.
7. Specify the trim settings (figure 2.78) and click on **Next**.
8. Finally, select a location to save outputs to and click on **Finish**.

Workflow tools and outputs

The Find QIAseq xHYB AMR Markers (Human host) template workflow consists of the following tools. See figure 2.79 for a full overview of the workflow.

- **QC for Sequencing Reads**. Performs basic quality control of the sequencing reads. The output, which is included in a combined report, can be used to evaluate the quality of the sequencing reads. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Sequencing_Reads.html.



Select which reference data set to use

○ Use the default reference data

● Select a reference set to use

<enter search term> □ Only Downloaded

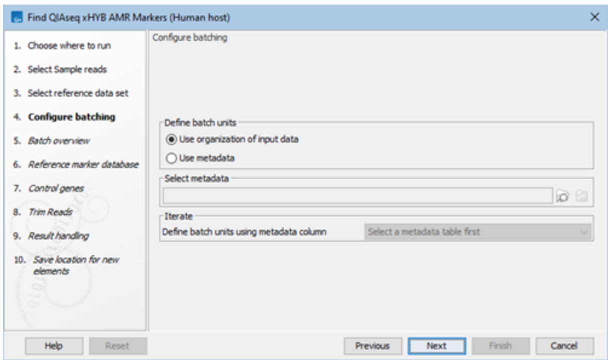
QIAGEN Active

QIAseq xHYB AMR Panel
QIAseq xHYB AMR Panel v1.0

Custom

The following types of reference data are used and must be supplied by the data set:
- control_genes
- peptide_marker_database

Figure 2.74: *Select reference data set.*



Configure batching

Define batch units

● Use organization of input data

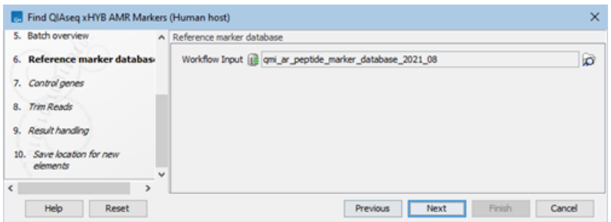
○ Use metadata

Select metadata

Iterate

Define batch units using metadata column: Select a metadata table first

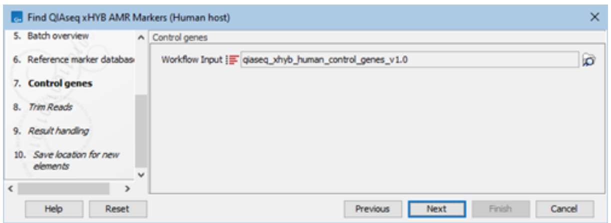
Figure 2.75: *Define batch units.*


Find QIAseq xHYB AMR Markers (Human host)
5. Batch overview
6. Reference marker databas
7. Control genes
8. Trim Reads
9. Result handling
10. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Reference marker database

Workflow Input: qmi_ar_peptide_marker_database_2021_08

Figure 2.76: *Select the reference marker database*

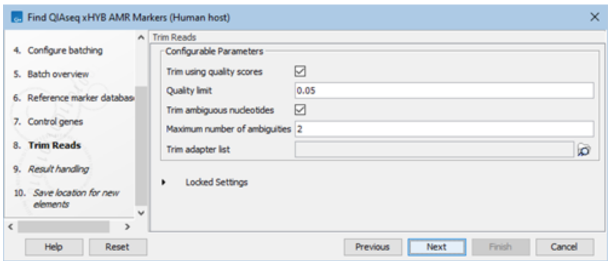

Find QIAseq xHYB AMR Markers (Human host)
5. Batch overview
6. Reference marker databas
7. Control genes
8. Trim Reads
9. Result handling
10. Save location for new elements
Help
Reset
Previous
Next
Finish
Cancel

Control genes

Workflow Input: qiaseq_xhyb_human_control_genes_v1.0

Figure 2.77: *Select control genes.*

- **Trim Reads.** Removes adapter sequences and low quality nucleotides. The appropriate settings for the Trim Reads tool depends on the protocol used to generate the reads. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_Reads.html.
- **Map Reads to Human Control Genes.** Maps the host reads output from **Taxonomic Profiling**



Find QIAseq xHYB AMR Markers (Human host)

4. Configure batching
5. Batch overview
6. Reference marker databas
7. Control genes
8. **Trim Reads**
9. *Result handling*
10. *Save location for new elements*

Trim Reads

Configurable Parameters

Trim using quality scores ☑

Quality limit 0.05

Trim ambiguous nucleotides ☑

Maximum number of ambiguities 2

Trim adapter list

▸ Locked Settings

Help | Reset | Previous | Next | Finish | Cancel

Figure 2.78: *Trim settings.*

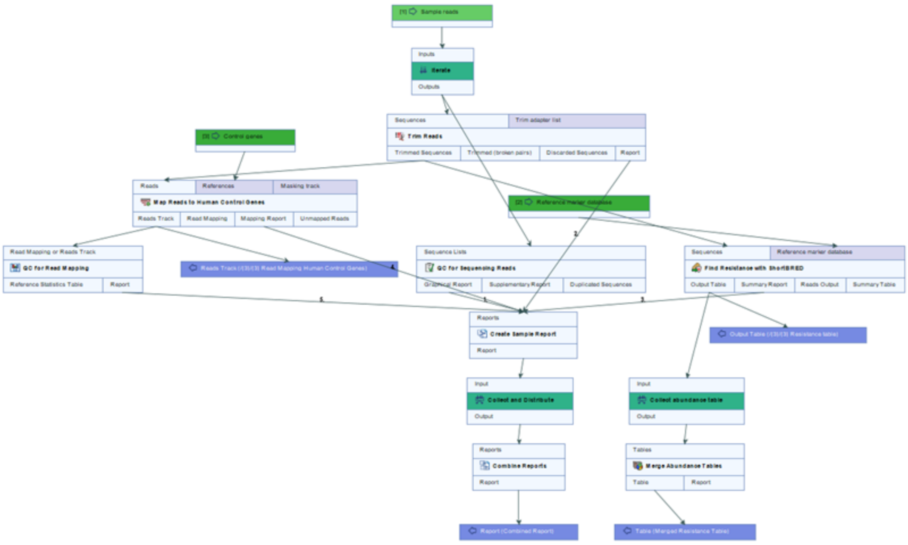


Figure 2.79: *Layout of the QIAseq xHYB AMR Markers (Human host) workflow.*

to the host taxonomic profiling index, to a reference of human control genes. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Map_Reads_Reference.html. This serves as a QC step to verify mapping to the human control genes. For human samples, you expect to see mapping of reads to all human control genes.

- **QC for Read Mapping**. Performs quality control of the read mapping. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QC_Read_Mapping.html.
- **Find Resistance with ShortBRED**. Detects and quantifies the presence of antimicrobial resistance marker genes of interest. See section 5.2.
- **Merge Abundance Tables**. Merges the sample-specific abundance tables to one combined abundance table. See section 6.1.

The outputs provided by this workflow are:

- **Resistance Table**. The result table from **Find Resistance with ShortBRED**. The sample-specific output provides the abundance of each detected AMR marker. See section 12.3.1.
- **Read Mapping Human Control Genes**. Sample-specific reads track containing the reads that mapped to the human control genes.

- **Combined Report**. Summary of the results from the individual tools for all the samples included in the workflow run.
- **Merged Resistance Table**. Provides the abundances of the detected AMR marker across samples. See section 12.3.1.

2.5 QIAseq Panel Analysis Assistant

The QIAseq Panel Analysis Assistant provides an easy entrance point for working with data generated with QIAseq panels and kits. Using the QIAseq Panel Analysis Assistant, information about the panels and kits can be accessed and available analyses can be viewed and run.

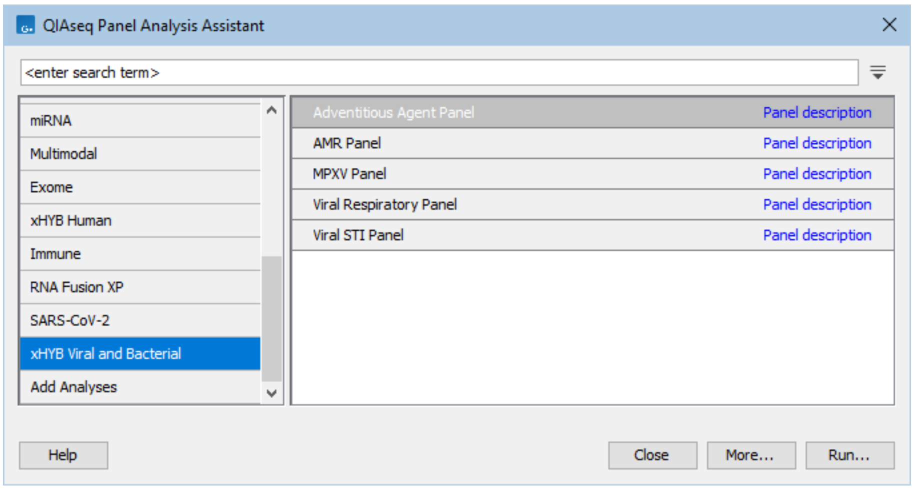
Most analyses offered via the QIAseq Panel Analysis Assistant are based on template workflows, which are available via the Toolbox. Analyses launched using the QIAseq Panel Analysis Assistant have the appropriate reference data preselected. Additionally, some parameters are different to the template workflow, to account for the panel/kit design.

Validation of results should be performed.

To start the QIAseq Panel Analysis Assistant, go to:

Toolbox | **Template Workflows** | **QIAseq Panel Analysis Assistant** (**)**

This opens a wizard listing different categories on the left, and analyses in the selected category on the right (figure 2.80).



Adventitious Agent Panel	Panel description
AMR Panel	Panel description
MPXV Panel	Panel description
Viral Respiratory Panel	Panel description
Viral STI Panel	Panel description

Figure 2.80: *The QIAseq Panel Analysis Assistant. Multiple analyses are available for the xHYB Viral and Bacterial category. The "Panel description" links to more information about the panel.*

An analysis can be:

- A pre-configured template workflow from the Toolbox.
- A pre-configured tool from the Toolbox.
- A tool only available from within the QIAseq Panel Analysis Assistant.

Once an analysis has been selected, it can be started using **Run**. Additional actions for the selected analysis are available under **More**.

For detailed information on the QIAseq Panel Analysis Assistant, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=QIAseq_Panel_Analysis_Assistant.html.

Part III

Metagenomics

Chapter 3

De Novo Assemble Metagenome

Before assembly, adapters should be removed from sequences. This can be done using **Trim Reads** (`https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_Reads.html`). The presence of adapters can result in the assembler trying to join regions that are not biologically relevant, leading to an assembly taking a long time and yielding misleading results.

Quality trimming before assembly is not generally necessary as the assembler itself should weed out or correct bad quality regions. However, trimming of low quality regions may decrease the amount of memory needed for the de novo assembly, which can be an advantage when working with large datasets.

To run the De Novo Assemble Metagenome tool:

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **De Novo Assemble Metagenome** ()

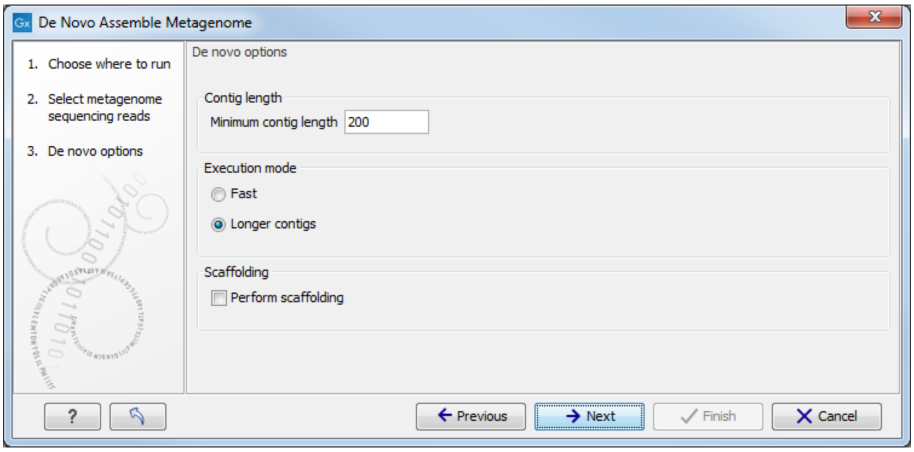
Select the sequence lists or single sequences to assemble.

Set assembly parameters (figure 3.1).

- **Minimum contig length** Contigs below this length will not be reported. For very complex datasets containing reads from many closely related species, the assembler will often produce shorter contigs. For such cases, it is recommended to set a lower threshold in order to cover a larger proportion of the metagenome with contigs. Reversely, for metagenomes of low complexity, it is often wise to set a higher threshold in order to avoid duplication.
- **Execution mode**
  - **Fast** The assembler is iterated once with a predifined wordsize ($k = 21$).
  - **Longer contigs** the assembler is iterated three times with increasing wordsize ($k = 21, 41, 61$), using the contigs from the previous iteration as input in the next iteration together with the input reads.

  Fast mode produces contigs of very high quality very fast, while the Longer contigs mode produces significantly longer contigs, possibly with slightly more misassemblies. Longer contigs mode requires up to three times more computation time.

- **Perform scaffolding** If selected, as the last step of the assembly process the assembler attempts to join contigs using paired-end information. Since paired-end information is needed to perform scaffolding, this option is disabled for single-end sequences.



De Novo Assemble Metagenome

1. Choose where to run
2. Select metagenome sequencing reads
3. De novo options

De novo options

Contig length

Minimum contig length 200

Execution mode

○ Fast

● Longer contigs

Scaffolding

□ Perform scaffolding

Figure 3.1: *Setting parameters for the assembly.*

Select **Create report** to create a summary report containing statistics on input reads and output contigs.

The De Novo Assemble Metagenome output

The De Novo Assemble Metagenome tool will output a list of contigs. If **Perform scaffolding** was selected, scaffolds will appear at the bottom of the contig list as scaffold_1, scaffold_2, etc.

Chapter 4

Amplicon-Based Analysis

In the Amplicon-Based Analysis folder you will find tools for analyzing amplicon data.

With **OTU Clustering** and accompanying OTU tools, you can cluster reads at e.g. 97% similarity, into so-called Operational Taxonomic Units (OTUs).

Detect Amplicon Sequence Variants offers a higher resolution alternative. It uses error profiling to distinguish biological nucleotide differences from sequencing errors.

Template workflows for amplicon-based analysis are available at:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Amplicon-Based Analysis ( )**

For more information on the template workflows, see section 2.2.

4.1 Normalize OTU Table by Copy Number

One way to correct OTU abundance tables is to take the rRNA copy number of the detected species into account and divide the detected read number for each OTU by the rRNA copy number. This can be done with the Normalize OTU Table by Copy Number (beta) tool. Note that this algorithm corrects the species distribution for each sample individually to get a more realistic picture of the species distribution in a sample. In order to normalize OTU abundance tables across samples one can use the Create Normalized Abundance Subtable button (see section 4.3.2), but many tools use an internal cross-sample normalization strategy.

In order to run this tool, an Amplicon Multiplication Table is required. Such tables can be imported with Import PICRUSt2 Multiplication Table (beta) 16.6.

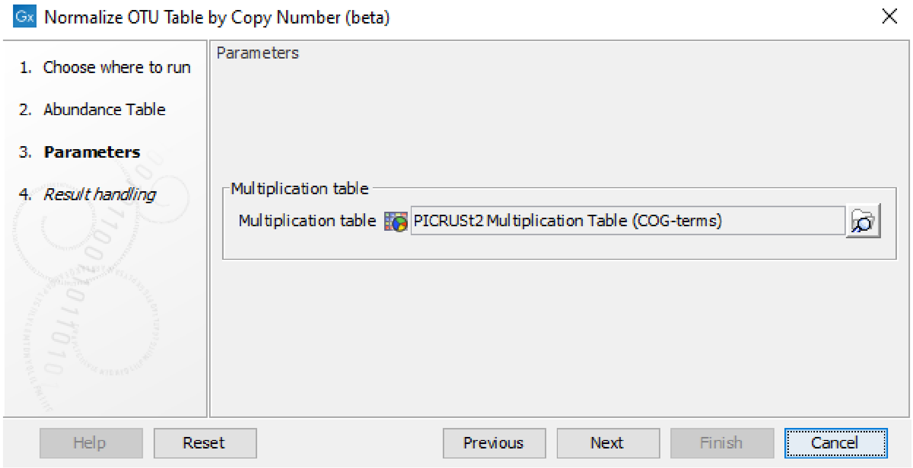
To run the tool, go to

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Amplicon-Based Analysis** ( ) | **Normalize OTU Table by Copy Number (beta)** ( )

Select the OTU table you want to normalize first and click "Next".

In the next wizard step (see figure 4.1) you can choose the Amplicon Multiplication Table to use.

The normalization works in the same way as the functional inference, except for that the functional



Normalize OTU Table by Copy Number (beta)

1. Choose where to run
2. Abundance Table
3. **Parameters**
4. *Result handling*

Parameters

Multiplication table

Multiplication table PICRUSt2 Multiplication Table (COG-terms)

Figure 4.1: *Selecting an Amplicon Multiplication Table for normalizing OTU tables.*

inference step is left out, see section 11.8.

4.2 Filter Samples Based on Number of Reads

In order to cluster accurately samples, they should have comparable coverage. Sometimes, however, DNA extraction, PCR amplification, library construction or sequencing has not been entirely successful, and a fraction of the resulting sequencing data will be represented by too few reads. These samples should be excluded from further analysis using the **Filter Samples Based on Number of Reads** tool.

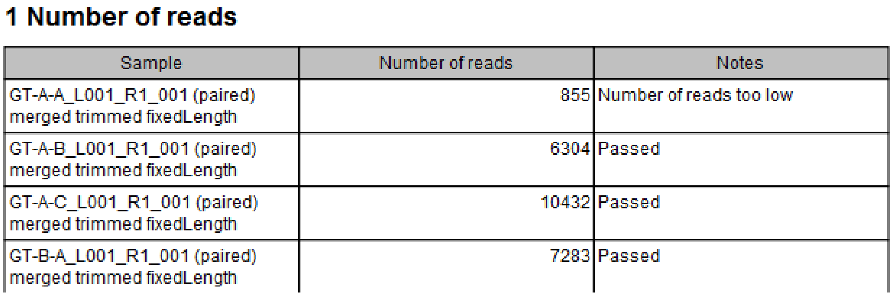
To run the tool, go to

Toolbox | **Microbial Genomics Module** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **Filter Samples Based on Number of Reads** ()

The tool requires that the input reads from each sample must be either all paired or all single. This check ensures that the samples are comparable, as the number of reads before merging paired reads is twice as great as the number of merged reads.

The threshold for determining whether a sample has sufficient coverage is specified by the parameters **minimum number of reads** and **minimum percent from the median**. The algorithm filters out all samples whose number of reads is less than the **minimum number of reads** or less than the **minimum percent from the median** times the median number of reads across all samples.

The primary output is a table describing how many reads are in a particular sample and if they passed or failed the quality control (see figure 4.2).



1 Number of reads

Sample	Number of reads	Notes
GT-A-A_L001_R1_001 (paired) merged trimmed fixedLength	855	Number of reads too low
GT-A-B_L001_R1_001 (paired) merged trimmed fixedLength	6304	Passed
GT-A-C_L001_R1_001 (paired) merged trimmed fixedLength	10432	Passed
GT-B-A_L001_R1_001 (paired) merged trimmed fixedLength	7283	Passed

Figure 4.2: *Output table from the Filter Samples Based on Number of Reads tool.*

In the next wizard window you can decide to **Copy samples with sufficient coverage** as well as to **Copy the discarded samples**. Copying the samples with sufficient coverage will give you a new list of sequences that you can use in your following analyses.

4.3 OTU clustering

The OTU clustering tool clusters a collection of reads to operational taxonomic units.

To run the tool, go to

Toolbox | **Microbial Genomics Module** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **OTU clustering** ()

The tool aligns the reads to reference OTU sequences (e.g. the reference database) to create an "alignment score" for each OTU. If the input sequence is shorter, the unaligned ends of the reference are ignored. For example, if a shorter sequence has 100% identity to a fragment of a longer reference sequence, the tool will assign 100% identity and assign the read to the OTU.

In the opposite case (longer read mapping to short database reference), the unaligned ends will count as indels, and the percentage identity will be lower.

When the input consists of paired reads, the OTU clustering tool will initially group them into pairs, and align both reads of a pair to the same OTUs. Both reads of a pair will be assigned to the one OTU where they BOTH align with the highest identity possible. Finally, the tool merges both reads of the pair using a stretch of N to the fragments so that the paired read looks as much as possible like the OTU they have been assigned to. For example, the forward-reverse pair (ACGACGACG, GTAGTAGTA) will be turned into ACGACGACGnnnnnnnnnnnnnnnnnnnTACTACTAC. Reads that cannot be merged will be independently aligned to reference OTUs.

If a read due to insufficient similarity cannot be included in an already existing OTU, the algorithm attempts to optimize the alignment score by allowing "crossover" from one database reference to another at a cost (the chimera crossover cost). To speed up the chimera crossover detection algorithm, the read is not aligned to all OTUs but only to the most promising candidates found via a k-mer search. If the best match has at least one crossover and the "constructed alignment" meets the similarity percentage threshold, the read is considered chimeric.

By default, the similarity percentage parameter is set to 97% in the OTU Clustering tool. Therefore without the chimera crossover cost, the constructed alignments difference score can only be 3% at most. The smaller the chimeric cost, the more likely it is that a read is deemed chimeric; setting it too high decreases the chimeric detection.

To add samples to existing OTU clustering results, we recommend to run OTU clustering on the new samples separately and use the tool Merge Abundance Tables described in section section 6.1 to merge the OTU tables. If re-running analysis is necessary and you wish to compare with previous results, you should keep the original sample input order. Due to the iterative nature of the clustering algorithm, changing the order of input files can lead to slightly different results. In most conceivable cases that difference does not matter, specifically when using taxonomy informed and abundance weighted distance metrics like weighted UniFrac.

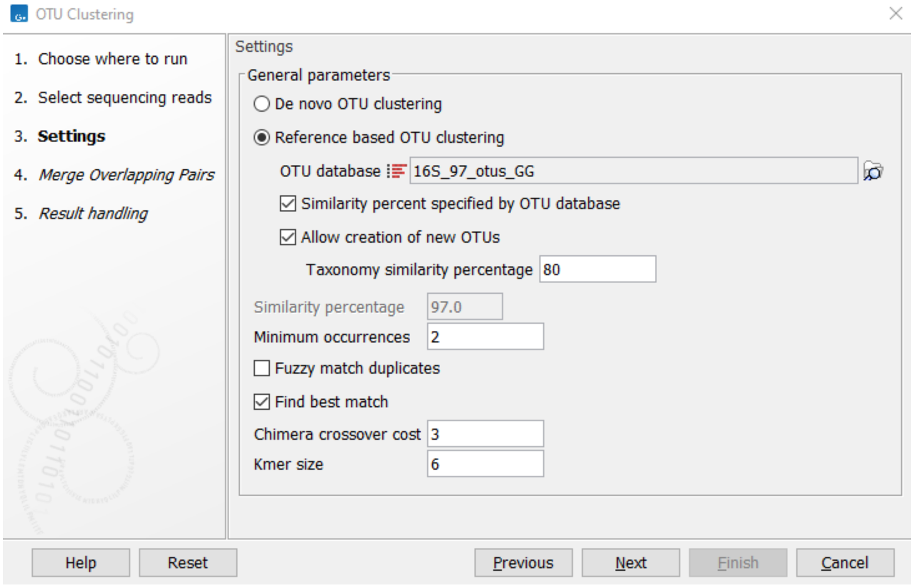
4.3.1 OTU clustering parameters

After having selected the sequences you would like to cluster, the wizard offers to set some general parameters (see figure 4.3).

You can choose to perform a **De novo OTU clustering**, or you can perform a **Reference based OTU clustering**.

The following parameters can be set:

- **OTU database** Specify the reference database to be used for Reference based OTU clustering. Reference databases can be created by the **Download Amplicon-Based Reference Database** tool or the **Update Sequence Attributes in Lists** tool (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Update_Sequence_Attributes_in_Lists.html).
- **Similarity percent specified by OTU database** Will apply the same similarity percentage (see below) as what was used when creating the reference database. This parameter is available only when performing a reference based OTU clustering. Selecting this parameter will disable the similarity percent parameter.



OTU Clustering

1. Choose where to run
2. Select sequencing reads
3. **Settings**
4. *Merge Overlapping Pairs*
5. *Result handling*

Settings

General parameters

○ De novo OTU clustering

● Reference based OTU clustering

OTU database: 16S_97_otus_GG

☑ Similarity percent specified by OTU database

☑ Allow creation of new OTUs

Taxonomy similarity percentage: 80

Similarity percentage: 97.0

Minimum occurrences: 2

□ Fuzzy match duplicates

☑ Find best match

Chimera crossover cost: 3

Kmer size: 6

Figure 4.3: *Settings for the OTU clustering tool.*

- **Allow creation of new OTUs** Allows sequences which are not already represented at the given similarity distance in the database to form a new cluster, and a new centroid is chosen. This parameter can be set only when performing a "Reference-based OTU clustering". Disallowing the creation of new OTUs is also known as closed reference OTU picking. Note that for input data where reads do not have the same orientation, the direction of the new OTUs cannot be inferred consistently. This may cause problems in downstream analyses (e.g. for estimating phylogenetic diversity).

- **Taxonomy similarity percentage** Specifies the similarity percentage to be used when annotating new OTUs. This parameter is available only when **Allow creation of new OTUs** is selected.

- **Similarity percentage** Specifies the required percentage of identity between a read and the centroid of an OTU for the read to join the OTU cluster.

- **Minimum occurrences** Specifies the minimum number of times a sequence must be represented in the read set for it to be included in the analysis. A value of 2 means that at least two reads representing a given sequence (i.e. duplicates) must be present for that sequence to be represented in further analysis. This option can be useful for filtering out singletons.

- **Fuzzy match duplicates** Specifies how to define duplicate reads. When not selected, reads that are 100% identical are considered duplicates. When selected, reads with 2% or fewer single nucleotide differences between them, and no other differences, are considered duplicates. The reads are sorted lexicographically (dictionary order) and then processed from most abundant to the least. Using this option, two or more singleton reads that are very similar may be marked as duplicates, allowing them to be included in further processing if together, their number exceeds the "Minimum occurrences" value.

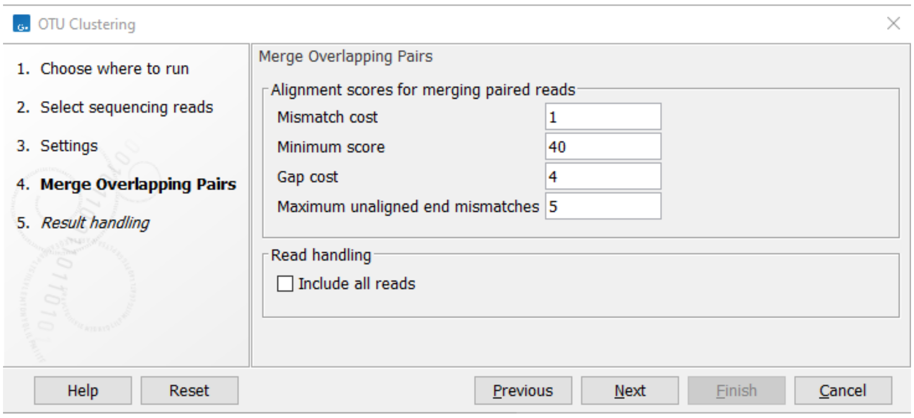
- **Find best match** If not selected, a read becomes a member of the first OTU-database entry found within the specified threshold. If the option is selected all database entries are tested and the read becomes a member of the best matching result. Note that "first" and "all" are relative terms in this case as kmer-searches are used to speed up the process. "All" only includes the database entries that the kmer search deems close enough, i.e., database entries that cannot be within the specified threshold will be filtered out at this step. "First" is the first matching entry as returned by the kmer-search which will sort by the number of kmer-matches.

- **Chimera crossover cost** The cost of doing a chimeric crossover, i.e. the higher the cost the less likely it is that a read is marked as chimeric.
- **Kmer size**: The size of the kmer to use in regards to the kmer usage in finding the best match.

Chimera detection is performed as follows: The read being processed is split into fragments. Each fragment is then queried for matches against the database with a k-mer search. Database references that match at least one query fragment are then selected and the read is then aligned to each selected reference while allowing "crossovers". Chimera detection is performed in order to identify any chimeric sequences, i.e., amplicons formed by joining two sequences during PCR. These are artifacts that will be excluded from the regular OTU clustering, and presented in a different abundance table labeled as being chimera-specific.

In order to use the highest quality sequences for clustering, it is recommended to merge paired read data. If the read length is smaller than the amplicon size, forward and reverse reads are expected to overlap in most of their 3' regions. Therefore, one can merge the forward and reverse reads to yield one high quality representative according to some pre-selected merge parameters: the overlap region and the quality of the sequences. For example, for a designed 150 bp overlap, a maximum score of 150 is achievable, but as the real length of the overlap is unknown, a lower minimum score should be chosen. Also, some mismatches and indels should be allowed, especially if the sequence quality is not perfect. You can also set penalties for mismatch, gap and unaligned ends.

In the Merge Overlapping Pairs dialog, you can set the parameters as seen in figure 4.4.


OTU Clustering

1. Choose where to run
2. Select sequencing reads
3. Settings
4. **Merge Overlapping Pairs**
5. *Result handling*

Merge Overlapping Pairs

Alignment scores for merging paired reads

Mismatch cost	1
Minimum score	40
Gap cost	4
Maximum unaligned end mismatches	5

Read handling

□ Include all reads

Figure 4.4: *Alignment parameters.*

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 1.
- **Minimum score** This is the minimum score of an alignment to be accepted for merging. The default value is 40. As an example: with default settings, this means that an overlap of 43 bases with one mismatch will be accepted (42 matches minus 1 for a mismatch).
- **Gap cost** This is the cost for introducing an insertion or deletion in the alignment. The default value is 4.
- **Maximum 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 (the default value is 5). However, this should be used with great care: 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. 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.

The tool accepts both paired and unpaired reads but will only merge paired reads in forward-reverse orientation. After merging, the merged reads will always be in the forward orientation.

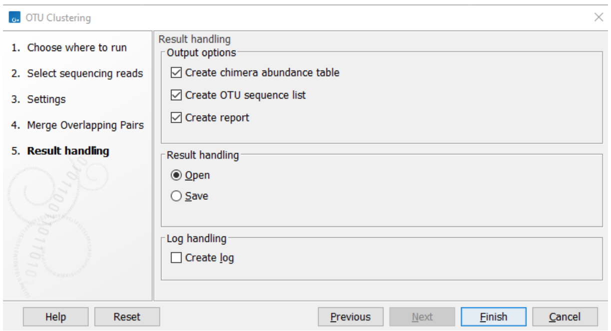
- **Include all reads** Select this option to include the non-merged reads in the OTU clustering analysis. If some or most of your paired reads are not expected to overlap, you should check this option to include all reads in the analysis. An example of an application resulting in non-overlapping paired reads would be fungal ITS sequencing, where you often sequence a larger amplicon than what can be covered by your read pairs.

4.3.2 OTU clustering outputs

Click **Next** to select outputs (figure 4.5).

In addition to the OTU abundance table, the following outputs are available:

- A sequence list of the OTUs
- A chimera abundance table with abundances for chimeras in each sample.
- A report that summarizes the results of the OTU clustering. For paired-end data, the report will include a section about the merging of overlapping paired reads.



Result handling

Output options

☑ Create chimera abundance table

☑ Create OTU sequence list

☑ Create report

Result handling

● Open

○ Save

Log handling

□ Create log

Figure 4.5: *OTU Clustering output options*

The OTU report

An example of an OTU report is shown in figure 4.6. The report contains the following sections:

- **OTU clustering**
  - **Input database size** The number of sequences in the input OTU database.
  - **Filtered database size** The number of sequences in the input OTU database having input reads mapped to it.
  - **OTUs based on database** The number of OTUs based on a sequence from the database.
  - **De novo OTUs** The number of OTUs not based on a sequence from the database.
  - **Total predicted OTUs** The total number of OTUs found.
- **Reads**
  - **Number of reads** The number of input reads
  - **Filtered reads** The number of reads filtered due to the minimum occurrences parameter. When reads are not at a specified similarity distance with the database, and the option to create new OTUs is not selected, these reads will be filtered as well.
  - **Unique reads after filtering** The number of unique reads after filtering. This is the number of candidates for OTUs before clustering.
  - **Chimeric reads** The number of reads detected as chimeric during clustering.
  - **Unique chimeric reads** The number of unique reads detected as chimeric.
  - **Reads in OTUs** The number of reads that contribute to the output OTUs.
- **Sample details**
  - **Sample** The name of the sample for which the following details are shown.
  - **Total number of reads** The number of input reads from the given sample.
  - **Filtered or chimeric reads** The number of reads from the given sample that were filtered due to the minimum occurrences parameter or detected as chimeric during clustering.
  - **Reads in OTUs** The number of reads from the given sample that contribute to the output OTUs.

- **Merging of paired reads** the following is reported for each input sample (generated if the input reads were paired)
    - **Summary** The number of merged, not merged and total paired reads.
    - **Merged pairs length distribution** Distribution of the lengths of the read pairs with the length of a read in base pairs on the x-axis and on the y-axis in the number of times a read of a given lengths has been observed.

The OTU abundance table

The OTU abundance table contains a list of OTUs, per-sample abundance values, and total abundance counts. Note that if the input contains paired-end sequences, each pair is counted as one read. There are a number of ways to visualize the contents of an OTU abundance table:

- **Table view** (▦) (figure 4.7)

  The table displays the following columns:

    - **Name** The name of the OTU, specified by either the reference database or by the OTU representative (see below for more details).
    - **Taxonomy** The taxonomy of the OTU, as specified by the reference database when a database entry was used as Reference.
    - **Combined Abundance** The total number of reads belonging to the OTU across all samples.
    - **Min** Minimum abundance across all samples
    - **Max** Maximum abundance across all samples
    - **Mean** Mean abundance of all samples
    - **Median** Median abundance of all samples
    - **Std** Standard deviation of all samples
    - **Abundance for each sample** The number of reads belonging to the OTU in a specific sample.
    - **Sequence** The sequence of the centroid of the OTU.

  Note on OTU Names: The name is either

    - The OTU name in the reference database (e.g. 978664)
    - The name of the read used as centroid, which for sequencing data may look like random numbers and letters. If the same name is present more than once, then the OTUs will have a trailing number "-00123" like readName-12345.
    - If there is no name (for new clusters where reads have no name), something like OTU-12345 is assigned.

  This will occur when one chooses the option "De novo OTU clustering" in the General parameters section of the OTU Clustering wizard, or the option "Allow creation of new OTUs". When either of these options are selected, it will be possible for the OTU clustering tool to create representative OTU sequences that are not in an existing reference database.

  In the right side panel, under the tab Data, you can switch between absolute counts and relative abundances (relative abundances are computed as the ratio between the number of reads belonging to the OTU in a specific sample and the total number of reads in the sample). You can also combine absolute counts and relative abundances by taxonomic levels by selecting the appropriate phylum in the **Aggregate feature** drop-down menu. Use the option below to Hide samples for which the taxonomy at the aggregated taxonomic level is incomplete. Finally, if you have previously annotated your table with Metadata (see section 6.8), you can **Aggregate sample** by the groups previously defined in your metadata table. This is useful when analyzing replicates from the same sample origin.

Under the table, the following actions are available:

- **Create Abundance Subtable** will create a table containing only the selected rows.
- **Create Sequence Sublist** will create a sequence list containing only the selected rows.
- **Create Normalized Abundance Subtable** will create a table with all rows normalized on the values of a single selected row. The row used for normalization will disappear from the new abundance table. The normalization scales the abundance table linearly, where the scaling factor is calculated by determining the average abundance across all samples and for each sample scale it to the average for the reference. Note that to be enabled, the selected row for normalization can only have non null abundance values. If you have zero values in some samples for the control, you will need to generate a new abundance table where these samples are not present. If the abundance table is obtained from merging single-sample abundance tables, then the merge should be redone excluding the samples with zero control read counts.

- **Stacked visualization view** (▦)

  In the Stacked Bar (figure 4.8) and Stacked Area Charts (figure 4.9), the metadata can be used to aggregate groups of columns (samples) by selecting the relevant metadata category in the right hand side panel. Also, the data can be aggregated at any taxonomy level selected. The relevant data points will automatically be summed accordingly.

  Holding the pointer over a colored area in any of the plots will result in the display of the corresponding taxonomy label and counts. With **Filter level** you can modify the number of features shown in the plot. For example, setting the value to 10 means that the 10 most abundant features of each sample will be shown in all columns. The remaining features are grouped into "Other", and will be shown if the option is selected in the right hand side panel. One can select which taxonomy level to color, and change the default colors manually. Colors can be be specified at the same taxonomy level as the one used to aggregate the data or at a lower level. When lower taxonomy levels are chosen in the data aggregation field, the color will be inherited in alternating shadings. It is also possible to sort samples by metadata attributes, and to show groups of samples without collapsing their stacks, as well as change the label of each stack or group of stacks. Features can be sorted by "abundance" or "name" using the drop down menu in the right hand side panel. Using the bottom right-most button (**Save/restore settings** (☰)), the settings can be saved and applied in other plots, allowing visual comparisons across analyses.

- **The sunburst view** (◎)

  The zoomable sunburst view lets the user select how many taxonomy level counts to display, and which level to color. Lower levels will inherit the color in alternating shadings. Taxonomy and relative abundances (the ratio between the number of reads belonging to the OTU in a specific sample and the total number of reads in the sample) are displayed in a legend to the left of the plot when hovering over the sunburst viewer with the mouse. The metadata can be used to select which sample or group of samples to show in the sunburst (figure 4.20).

Clicking on a lower level field will render that field the center of the plot and display lower level counts in a radial view. Clicking on the center field will render the level above the current view the center of the view (figure 4.11).

4.3.3 Importing and exporting OTU abundance tables

It is possible to import a biom, a csv or an excel file as an OTU abundance table, by going to **File** | **Import** () | **Standard Import...** () and force the input as type "OTU abundance table (.xls, .xlsx, .csv)" or "Biom (.biom)". Currently supported versions for BIOM file format are versions 1.0 and 2.1.

This importer allows users to perform statistical analyses on abundance tables that were not generated by OTU clustering tool. Note that abundance tables that are imported will not contain metadata or grouping information, and thus metadata has to be re-applied using the Add Metadata to Abundance Table tool after import.

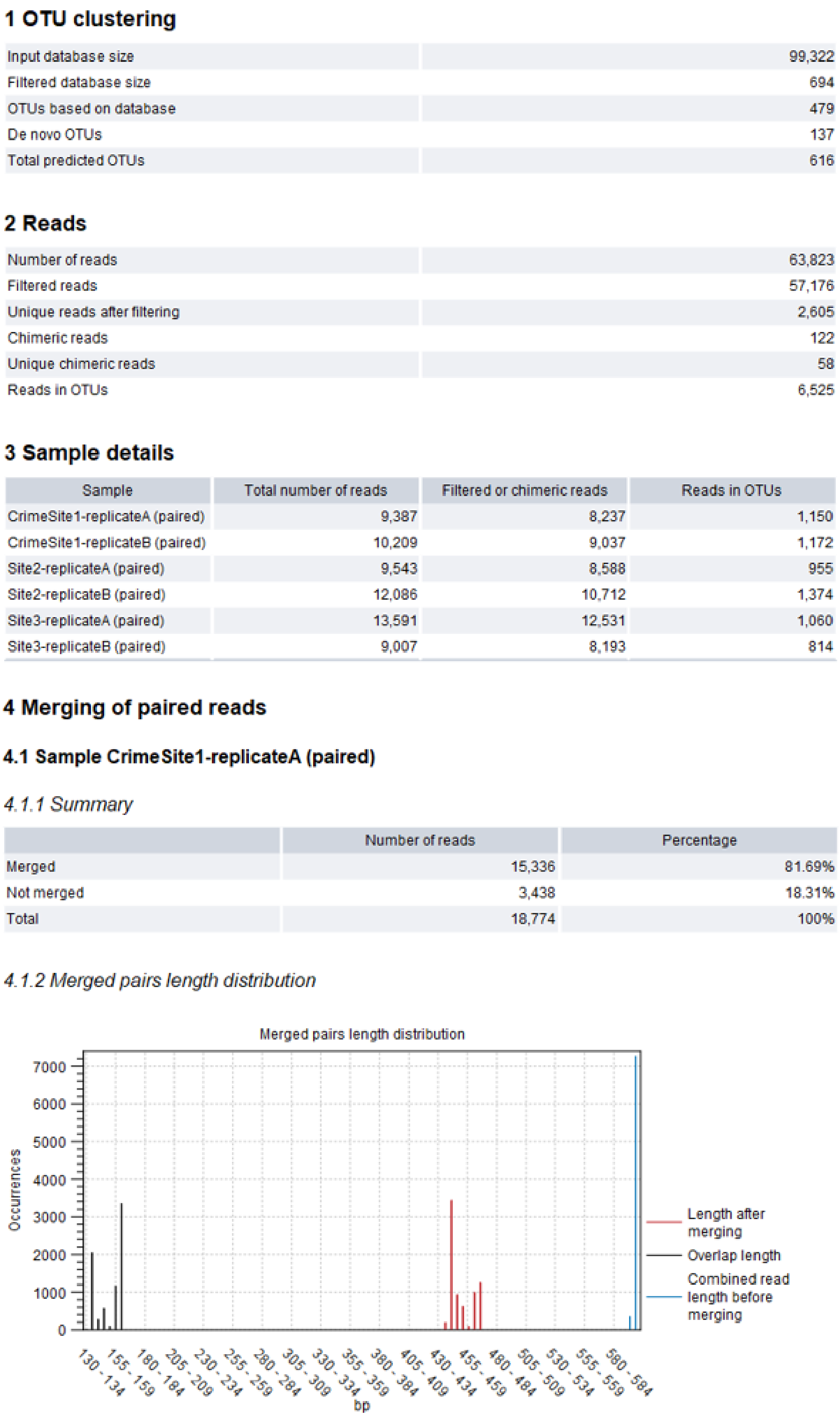
For example, Terminal Restriction Fragment Length Polymorphism (TRFLP) data can be imported and treated similarly as OTU abundance tables. However, all sequence-based actions cannot be applied to this data (i.e., multiple sequence alignment, tree reconstruction and phylogenetic tree measure estimation).

The importer recognizes the following column headers:

- **Name** The name of the OTU, specified by either the reference database or by the OTU representative.
- **Taxonomy** The taxonomy of the OTU, as specified by the reference database when a database entry was used as reference, e.g "Bacteria; Bacillota; Bacilli; Lactobacillales; Lactobacillaceae; Lactobacillus; Lactobacillus gasseri".
- **Sequence** The sequence of the centroid of the OTU.
- Any other header of a column with integer values: The header is interpreted as sample name and the values as abundance values. Values must be absolute counts and not relative abundances.

It is furthermore possible to export abundance tables to different formats, but it is recommended to use the Biological Observation Matrix (biom) file format (biom-format.org) as a standardized format. Currently, the only supported version for export is 2.1.

Sunbursts graphs can be exported in the following formats: *.jpg, *.tif, *.png, *.ps, *.eps, *.svg.



1 OTU clustering

Input database size	99,322
Filtered database size	694
OTUs based on database	479
De novo OTUs	137
Total predicted OTUs	616

2 Reads

Number of reads	63,823
Filtered reads	57,176
Unique reads after filtering	2,605
Chimeric reads	122
Unique chimeric reads	58
Reads in OTUs	6,525

3 Sample details

Sample	Total number of reads	Filtered or chimeric reads	Reads in OTUs
CrimeSite1-replicateA (paired)	9,387	8,237	1,150
CrimeSite1-replicateB (paired)	10,209	9,037	1,172
Site2-replicateA (paired)	9,543	8,588	955
Site2-replicateB (paired)	12,086	10,712	1,374
Site3-replicateA (paired)	13,591	12,531	1,060
Site3-replicateB (paired)	9,007	8,193	814

4 Merging of paired reads

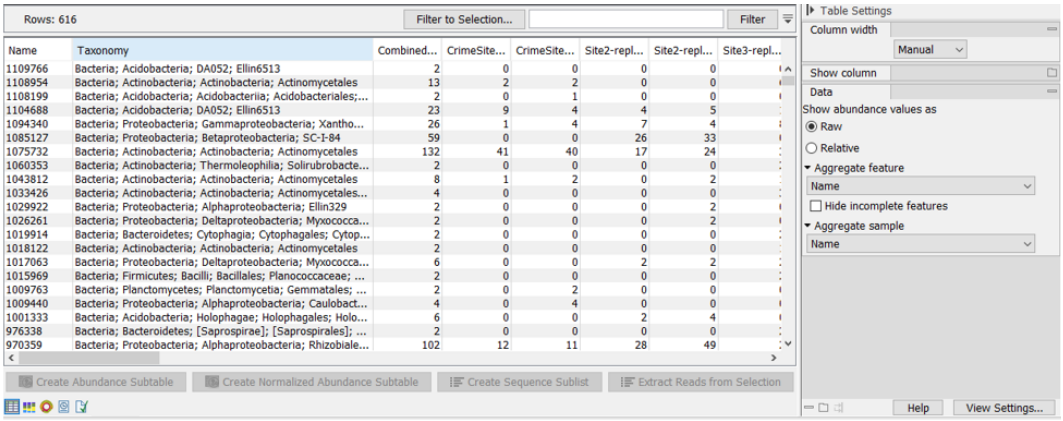
4.1 Sample CrimeSite1-replicateA (paired)

4.1.1 Summary

	Number of reads	Percentage
Merged	15,336	81.69%
Not merged	3,438	18.31%
Total	18,774	100%

4.1.2 Merged pairs length distribution

Figure 4.6: *Example of report produced by the OTU clustering tool.*



Name	Taxonomy	Combined...	CrimeSite...	CrimeSite...	Site2-repl...	Site2-repl...	Site3-repl...
1109766	Bacteria; Acidobacteria; DA052; Ellin6513	2	0	0	0	0	
1108954	Bacteria; Actinobacteria; Actinobacteria; Actinomycetales	13	2	2	0	0	
1108199	Bacteria; Acidobacteria; Acidobacteriia; Acidobacteriales;...	2	0	1	0	0	
1104688	Bacteria; Acidobacteria; DA052; Ellin6513	23	9	4	4	5	
1094340	Bacteria; Proteobacteria; Gammaproteobacteria; Xantho...	26	1	4	7	4	
1085127	Bacteria; Proteobacteria; Betaproteobacteria; SC-I-84	59	0	0	26	33	
1075732	Bacteria; Actinobacteria; Actinobacteria; Actinomycetales	132	41	40	17	24	
1060353	Bacteria; Actinobacteria; Thermoleophilia; Solirubrobacte...	2	0	0	0	0	
1043812	Bacteria; Actinobacteria; Actinobacteria; Actinomycetales	8	1	2	0	2	
1033426	Bacteria; Actinobacteria; Actinobacteria; Actinomycetales...	4	0	0	0	0	
1029922	Bacteria; Proteobacteria; Alphaproteobacteria; Ellin329	2	0	0	0	2	
1026261	Bacteria; Proteobacteria; Deltaproteobacteria; Myxococca...	2	0	0	0	2	
1019914	Bacteria; Bacteroidetes; Cytophagia; Cytophagales; Cytop...	2	0	0	0	0	
1018122	Bacteria; Actinobacteria; Actinobacteria; Actinomycetales	2	0	0	0	0	
1017063	Bacteria; Proteobacteria; Deltaproteobacteria; Myxococca...	6	0	0	2	2	
1015969	Bacteria; Firmicutes; Bacilli; Bacillales; Planococcaceae; ...	2	0	0	0	0	
1009763	Bacteria; Planctomycetes; Planctomycetia; Gemmatales; ...	2	0	2	0	0	
1009440	Bacteria; Proteobacteria; Alphaproteobacteria; Caulobact...	4	0	4	0	0	
1001333	Bacteria; Acidobacteria; Holophagae; Holophagales; Holo...	6	0	0	2	4	
976338	Bacteria; Bacteroidetes; [Saprospirae]; [Saprospirales]; ...	2	0	0	0	0	
970359	Bacteria; Proteobacteria; Alphaproteobacteria; Rhizobiale...	102	12	11	28	49	

Column width: Manual

Show column

Data

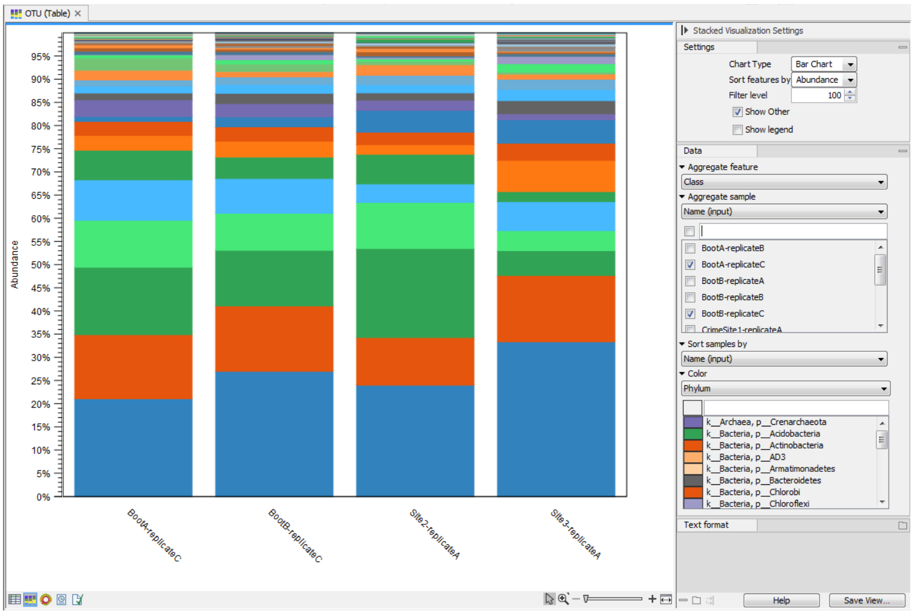
Show abundance values as
● Raw
○ Relative

Aggregate feature: Name

□ Hide incomplete features

Aggregate sample: Name

Figure 4.7: *OTU abundance table.*



Settings

Chart Type: Bar Chart
Sort features by: Abundance
Filter level: 100
☑ Show Other
□ Show legend

Data

Aggregate feature: Class

Aggregate sample: Name (input)

□ BootA-replicateB
☑ BootA-replicateC
□ BootB-replicateA
□ BootB-replicateB
☑ BootB-replicateC
□ CrimeSite1-replicateA

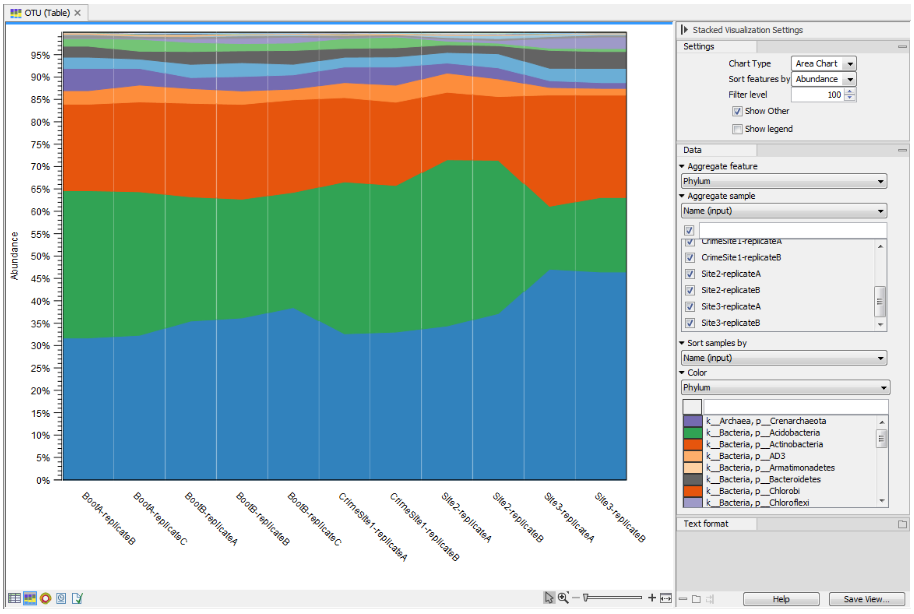
Sort samples by: Name (input)

Color: Phylum

🟪 k__Archaea, p__Crenarchaeota
🟩 k__Bacteria, p__Acidobacteria
🟥 k__Bacteria, p__Actinobacteria
🟧 k__Bacteria, p__AD3
🟧 k__Bacteria, p__Armatimonadetes
⬛ k__Bacteria, p__Bacteroidetes
🟥 k__Bacteria, p__Chlorobi
🟪 k__Bacteria, p__Chloroflexi

Text format

Figure 4.8: *Stacked bar of the microbial community at the class level for 4 different samples.*



Settings

Chart Type: Area Chart
Sort features by: Abundance
Filter level: 100
☑ Show Other
□ Show legend

Data

Aggregate feature: Phylum

Aggregate sample: Name (input)

☑ CrimeSite1-replicateA
☑ CrimeSite1-replicateB
☑ Site2-replicateA
☑ Site2-replicateB
☑ Site3-replicateA
☑ Site3-replicateB

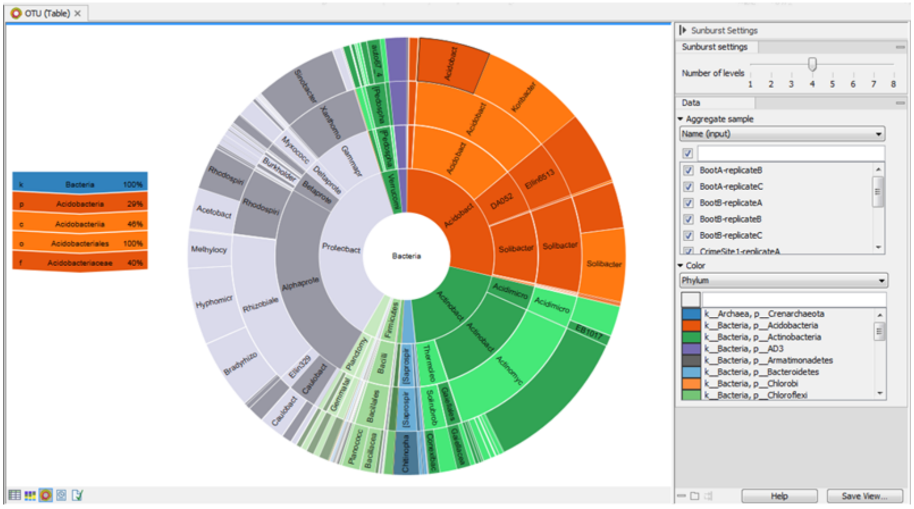
Sort samples by: Name (input)

Color: Phylum

🟪 k__Archaea, p__Crenarchaeota
🟩 k__Bacteria, p__Acidobacteria
🟥 k__Bacteria, p__Actinobacteria
🟧 k__Bacteria, p__AD3
🟧 k__Bacteria, p__Armatimonadetes
⬛ k__Bacteria, p__Bacteroidetes
🟥 k__Bacteria, p__Chlorobi
🟪 k__Bacteria, p__Chloroflexi

Text format

Figure 4.9: *Stacked area of the microbial community at the phylum level for 11 different sites.*



k	Bacteria	100%
p	Acidobacteria	29%
c	Acidobacteriia	46%
o	Acidobacteriales	100%
f	Acidobacteriaceae	40%

Sunburst Settings

Sunburst settings

Number of levels: 1 2 3 4 5 6 7 8 (4)

Data

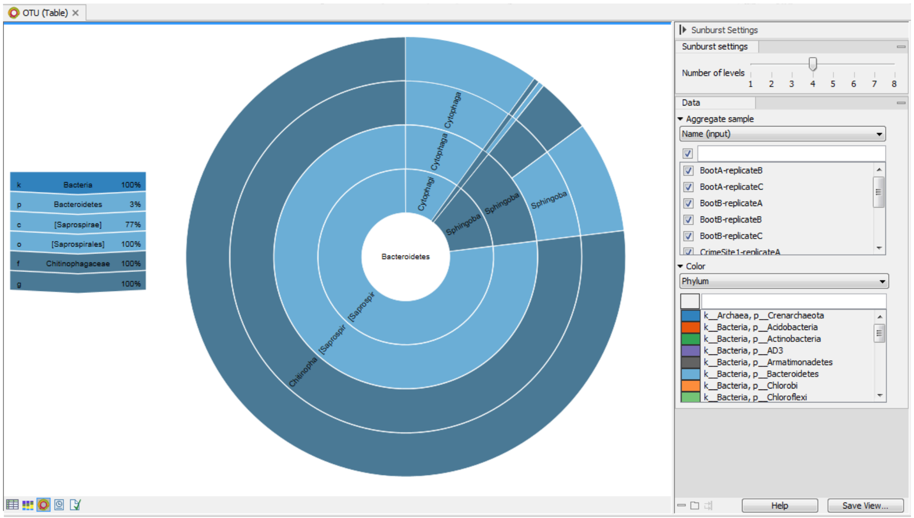
Aggregate sample: Name (input)

☑ BootA-replicateB
☑ BootA-replicateC
☑ BootB-replicateA
☑ BootB-replicateB
☑ BootB-replicateC
☑ CrimeSite1-replicateA

Color: Phylum

k__Archaea, p__Crenarchaeota
k__Bacteria, p__Acidobacteria
k__Bacteria, p__Actinobacteria
k__Bacteria, p__AD3
k__Bacteria, p__Armatimonadetes
k__Bacteria, p__Bacteroidetes
k__Bacteria, p__Chlorobi
k__Bacteria, p__Chloroflexi

Figure 4.10: *Sunburst view of the microbial community showing all taxa belonging to the kingdom bacteria.*



k	Bacteria	100%
p	Bacteroidetes	3%
c	[Saprospirae]	77%
o	[Saprospirales]	100%
f	Chitinophagacese	100%
g		100%

Sunburst Settings

Sunburst settings

Number of levels: 1 2 3 4 5 6 7 8 (4)

Data

Aggregate sample: Name (input)

☑ BootA-replicateB
☑ BootA-replicateC
☑ BootB-replicateA
☑ BootB-replicateB
☑ BootB-replicateC
☑ CrimeSite1-replicateA

Color: Phylum

k__Archaea, p__Crenarchaeota
k__Bacteria, p__Acidobacteria
k__Bacteria, p__Actinobacteria
k__Bacteria, p__AD3
k__Bacteria, p__Armatimonadetes
k__Bacteria, p__Bacteroidetes
k__Bacteria, p__Chlorobi
k__Bacteria, p__Chloroflexi

Figure 4.11: *Sunburst view of the microbial community zoomed to show all taxa belonging to the phylum Bacteroidetes.*

4.4 Remove OTUs with Low Abundance

Low abundance OTUs can eliminated from the OTU table if they have fewer than a given count across all the samples in the experiment.

To run the tool, go to

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Amplicon-Based. Analysis** () | **Remove OTUs with Low Abundance** ()

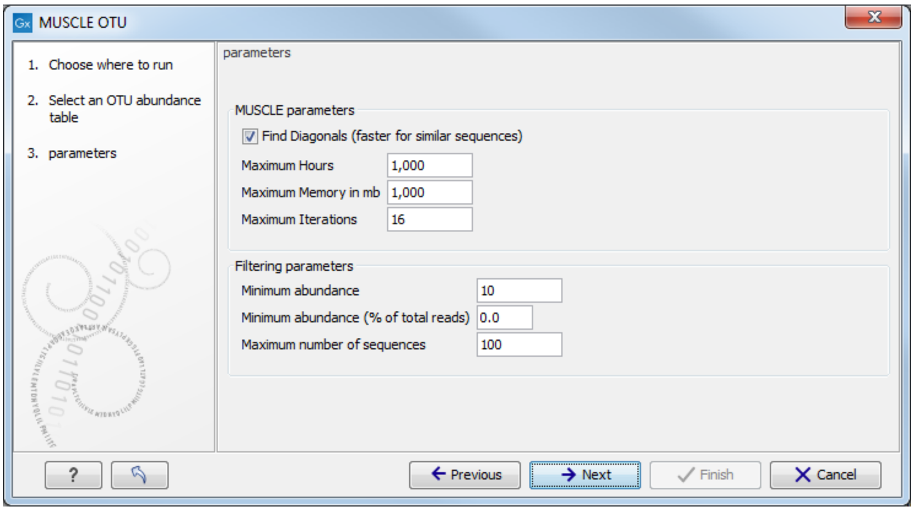
Choose an OTU table as input, select the filtering parameters and save the table. The threshold for determining whether an OTU has sufficient abundance is specified by the parameters **minimum combined abundance** and **minimum combined abundance (% of all the reads)**. The algorithm filters out all OTUs whose combined abundance across all samples is less than the minimum combined abundance or whose combined abundance is less than the minimum combined abundance (% of all the reads) across all samples. The default value for the Minimum combined abundance is set at 10.

4.5 Align OTUs with MUSCLE

To estimate Alpha and Beta diversity, OTUs must initially be aligned with the MUSCLE tool of the CLC Microbial Genomics Module:

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **Align OTUs using MUSCLE** ()

Choose an OTU abundance table as input. The next wizard window allows you to set up the alignment parameters with MUSCLE (figure 4.12).



MUSCLE OTU

1. Choose where to run
2. Select an OTU abundance table
3. parameters

parameters

MUSCLE parameters

☑ Find Diagonals (faster for similar sequences)

Maximum Hours: 1,000

Maximum Memory in mb: 1,000

Maximum Iterations: 16

Filtering parameters

Minimum abundance: 10

Minimum abundance (% of total reads): 0.0

Maximum number of sequences: 100

Figure 4.12: *Set up parameters for aligning sequences with MUSCLE.*

- **Find Diagonals**: you can decide on some restrictive parameters for your analysis: the **Maximum Hours** the analysis should last, the **Maximum Memory in mb** that should be used for the analysis, or the **Maximum Iterations** the analysis should make. The latter is set to 16 by default.

- **Filtering Parameters**: The algorithm filters out all OTUs whose combined abundance across all samples is less than the **minimum combined abundance** or whose combined abundance is less than the **minimum combined abundance (% of all the reads)** across all samples. The default value for the Minimum combined abundance is set at 10. Moreover, you can specify the **Maximum number of sequences to be aligned**, so that only the sequences with the highest combined abundances will be used. **Note** that reducing the number of sequences will speed up the alignment and the construction of phylogeny trees.

Note that by default only the top 100 most abundant OTUs are aligned using MUSCLE and used to reconstruct the phylogeny tree in the next step. This phylogenetic tree is used for calculating the phylogenetic diversity and the UniFrac distances, so these measures disregard the low abundance OTUs by default. If more OTUs are to be included, the default settings for the MUSCLE alignment need to be changed accordingly.

For further analysis with the Alpha and Beta diversity tools, save the alignment and construct a phylogenetic tree using the Maximum Likelihood Phylogeny tool (use the Launch button to find it in the Toolbox). For more information, see `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Maximum_Likelihood_Phylogeny.html`.

4.6 Detect Amplicon Sequence Variants

The Detect Amplicon Sequence Variants tool infers sequence variants from amplicon data. The tool uses error profiling to distinguish biological nucleotide differences from sequencing errors, making it possible to resolve amplicon sequence variants (ASVs) down to the level of single nucleotide differences. The algorithm is inspired by DADA2, [Callahan et al., 2016].

The Detect Amplicon Sequence Variants analysis includes the following steps:

- Initial filtering and length trimming ensures that reads are of the same length and optimized for the subsequent analysis:
  - **Length trimming** Reads are trimmed from the 3' end to the user-defined length. Reads shorter than this are removed.
  - **Ambiguity filter** Reads containing ambiguous bases are discarded.
  - **Expected Errors filter** Reads with more expected errors than the user-defined threshold are discarded.
- **Dereplication** Produces an intermediate list of unique sequences.
- **Denoising** This iterative process estimates a sample-specific error model. This error model is then used to distinguish biological nucleotide differences from likely sequencing errors and generate the list of candidate amplicon sequence variants.
- **Remove chimeras** Sequences that are assessed as being chimeras are discarded.
- **Merging unique read pairs** For paired read dataset, unique read pairs are merged. Pairs with insufficient overlap (<12 bases), are discarded.

A template workflow with a proposed analysis pipeline - trimming reads, detecting amplicon sequence variants, merging ASV tables, and assigning taxonomies - is available at:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **Detect Amplicon Sequence Variants and Assign Taxonomies workflow** ()

For more information, see section 2.2.2.

4.6.1 Detect Amplicon Sequence Variants parameters

To run the Detect Amplicon Sequence Variants tool, go to

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Amplicon-Based Analysis** () | **Detect Amplicon Sequence Variants** ()

Select the single- or paired-end sequence lists to be analyzed. For paired reads, pairs should have a minimum overlap of 12 bases. Data should be trimmed beforehand to remove adapters and poor quality nucleotides. This can be done using **Trim Reads** (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_Reads.html).

Set the Trim and filter parameters (figure 4.13):

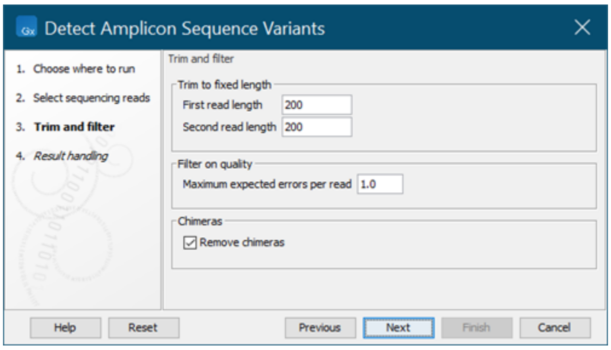
- **First/Second read length**: Reads are trimmed to the given length from the 3' end. Reads shorter than this will be discarded. For paired reads you may set different values for the first and second read in a pair.

  What values to set will depend on your sequencing protocol and how reads are trimmed prior to being used as input for the Detect Amplicon Sequence Variants tool. We recommend that you have a look at the Trim report section *Read length before / after trimming* if you are unsure about what value to set.

  You must use the same length setting for all samples that will be compared in downstream analysis.

- **Maximum expected errors per read**: The maximum number of expected errors allowed for a read. Reads with more expected errors will be discarded.

- **Remove chimeras**: If selected, reads identified as chimeras will be discarded.



Detect Amplicon Sequence Variants

1. Choose where to run
2. Select sequencing reads
3. **Trim and filter**
4. *Result handling*

Trim and filter

Trim to fixed length

First read length 200

Second read length 200

Filter on quality

Maximum expected errors per read 1.0

Chimeras

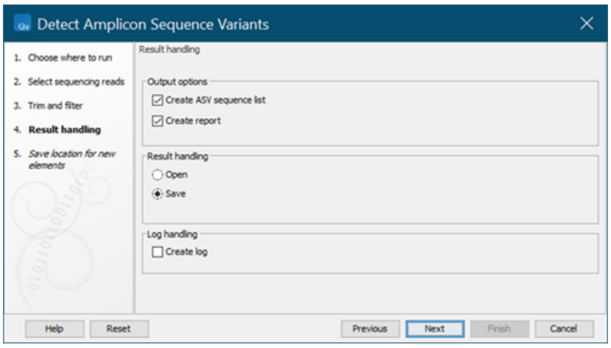
☑ Remove chimeras

Figure 4.13: *Trim and filter parameter settings*

4.6.2 Detect Amplicon Sequence Variants output

Click **Next** to select the output (figure 4.14).

In addition to an ASV abundance table, the following outputs are available:



Detect Amplicon Sequence Variants

1. Choose where to run
2. Select sequencing reads
3. Trim and filter
4. **Result handling**
5. *Save location for new elements*

Result handling

Output options

☑ Create ASV sequence list

☑ Create report

Result handling

○ Open

● Save

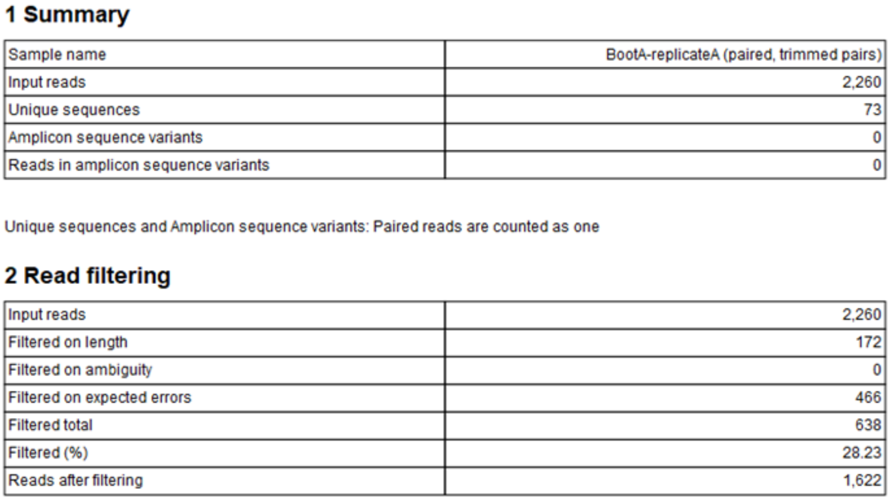
Log handling

□ Create log

Figure 4.14: *Detect Amplicon Sequence Variants output options*

- Click **Create ASV sequence list** to generate a sequence list with the detected amplicon sequence variants.
- Click **Create report** to generate a summary report.

The ASV report



1 Summary

Sample name	BootA-replicateA (paired, trimmed pairs)
Input reads	2,260
Unique sequences	73
Amplicon sequence variants	0
Reads in amplicon sequence variants	0

Unique sequences and Amplicon sequence variants: Paired reads are counted as one

2 Read filtering

Input reads	2,260
Filtered on length	172
Filtered on ambiguity	0
Filtered on expected errors	466
Filtered total	638
Filtered (%)	28.23
Reads after filtering	1,622

Figure 4.15: *First sections of the Detect Amplicon Sequence Variants report*

- **Summary** (figure 4.15)
  - **Sample name** The name of the sample.
  - **Input reads** The number of input reads.
  - **Unique sequences** The number of unique sequences detected in the input reads. Read pairs are counted as one.
  - **Amplicon sequences variants** The number of amplicon sequences variant detected in the input reads. Read pairs are counted as one.
  - **Reads in amplicon sequence variants** The number of reads grouped into ASVs.
- **Read filtering**
  - **Input reads** The number of input reads.

  - **Filtered on length** The number of reads that were removed because they were shorter than the defined length threshold.
  - **Filtered on ambiguity** The number of reads that were removed because they contained ambiguous bases.
  - **Filtered on expected errors** The number of reads that were removed because they exceeded the *Maximum expected errors* threshold.
  - **Filtered total** The total number of filtered reads.
  - **Filtered (%)** The percentage of reads filtered.
  - **Reads after filtering** The number of reads left after filtering.

- **Distribution of expected errors**
- **Read lengths** Plot of read lengths before and after length trimming and filtering.
- **Unique sequences**
- **Merging of unique read pairs**
  - **Number of unique pairs** The number of read pairs.
  - **Unique pair with insufficient overlap** The number of pairs that had insufficient overlap and were discarded.
  - **Merged unique pairs** The number of read pairs that were successfully merged.
- **Error model estimation**
  - **R1** Error model computed based on forward reads.
  - **R2** Error model computed based on reversed reads.

The ASV abundance table

The ASV abundance table contains the detected amplicon sequence variants (ASVs) and the abundance of each ASV. The Detect Amplicon Sequence Variants tool produces one ASV abundance table per sample. To go beyond single sample ASVs, you can combine tables and enrich them with metadata using the following tools:

- **Merge Abundance Tables** Creates a merged, multi-sample ASV abundance table that allows you to compare abundances across samples, see section 6.1.
- **Add Metadata to Abundance Table** Adds sample metadata to your table. This allows you to aggregate samples based on attributes. This is useful for instance when analyzing replicates from the same sample origin. See section 6.8 for information on how to add metadata.
- **Assign Taxonomies to Sequences in Abundance Table** Assigns taxonomy annotations to the ASVs. You can aggregate ASVs by taxonomy level, (see section 6.2).

In the following, we focus on the single sample ASV abundance table, but include a few hints about additional features and options that could be of use for **merged ASV abundance tables**,

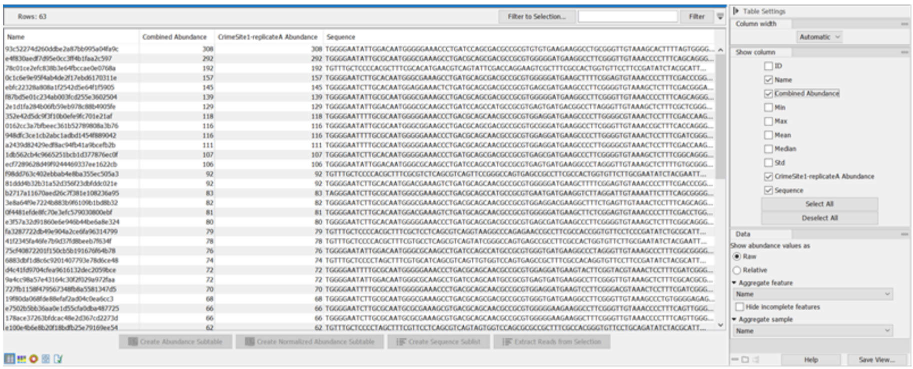
ASV abundance tables with sample metadata, and **ASV abundance tables with assigned taxonomies**.

There are a number of ways to visualize the ASV abundance table:

- **Table view** (▦) (figure 4.16) The table displays the following columns, some of which are of use mostly for merged, multi-sample ASV tables:
  - **ID** The ID of the ASV.
  - **Name** The name of the ASV. The name is generated as an MD5 hash ID, why identical ASVs will have the same name across ASV tables.
  - **Combined Abundance** The total number of reads belonging to the ASV across samples.
  - **Min** Minimum abundance across all samples
  - **Max** Maximum abundance across all samples
  - **Mean** Mean abundance of all samples
  - **Median** Median abundance of all samples
  - **Std** Standard deviation of all samples
  - **Abundance for each sample** The number of reads belonging to the ASV in a specific sample.
  - **Sequence** The sequence of the detected ASV.

  In the **Data** section of the **Side panel**, switch between *Raw* and *Relative* abundance. Relative abundance is computed as the ratio between the number of reads belonging to an ASV and the total number of reads in the sample.

  For **merged ASV abundance tables with sample metadata**, use the setting **Aggregate sample** to aggregate samples based on metadata attributes, e.g. replicates from the same sample origin.



Rows: 63

Name	Combined Abundance	CrimeSite1-replicateA Abundance	Sequence
93c52274d260ddbe2a87bb995a04fa9c	308	308	[illegible]
e4f830aedf7d95e0cc3ff4b1faa2c597	292	292	[illegible]
78c01ce2efc838b3e64fbccae0e0768a	192	192	[illegible]
0c1c6e9e95f4ab4de2f17ebd6170311e	157	157	[illegible]
ebfc22328a808a1f2542d5e64f1f5905	145	145	[illegible]
f87bd5e01c234ab003fcd255e3602504	139	139	[illegible]
2e1d1fa284b06fb59eb978c88b4905fe	129	129	[illegible]
352e42d5dc9f3f10b0efe9fc701e21af	118	118	[illegible]
0162cc3a7bfbeec361b527898808a3b76	116	116	[illegible]
948dfc3ce1cb2abc1adbd1454f889042	116	116	[illegible]
a2439d82429edf8ac94fb41a9bcefb2b	111	111	[illegible]
1db562cb4c9665251bcb1d377876ec0f	107	107	[illegible]
ecf7289628d49f9244469337ee1622cb	106	106	[illegible]
f98dd763c402ebbab4e8ba355ec505a3	92	92	[illegible]
81ddd4b32b31a52d356f23dbfddc021e	92	92	[illegible]
b2717a11670aed26c7f381e108236a95	83	83	[illegible]
3e8a64f9e7224b883b9f6109b1bd8b32	82	82	[illegible]
0f4481efde8fc70e3efc579030800ebf	81	81	[illegible]
e3f57a32d91860e6e946b44be6a8e324	80	80	[illegible]
fa3287722db49e904a2ce6fa96314799	79	79	[illegible]
41f2345fa46fe7b9d37fd8beeb7f634f	78	78	[illegible]
75cf40872201f150cb5b19167 6f64b78	76	76	[illegible]
6883dbf1d8c6c9201407793e78d6ce48	74	74	[illegible]
d4c41fd9704cfea9616132dec2059bce	72	72	[illegible]
9a4cc98a57e43164c30f2f029a972faa	72	72	[illegible]
727fb1158f479567348fb8a5581347d5	70	70	[illegible]
19f80da068fde88efaf2ad04c0ea6cc3	68	68	[illegible]
e7502b5bb36aa0e1d55cfa0dba487725	66	66	[illegible]
178ace37263bfdcac48e2d367cd2273d	66	66	[illegible]
e100e4b6e8b20f18bdfb25e79169ee54	62	62	[illegible]

Table Settings

Column width: Automatic

Show column

- ☐ ID
- ☑ Name
- ☑ Combined Abundance
- ☐ Min
- ☐ Max
- ☐ Mean
- ☐ Median
- ☐ Std
- ☑ CrimeSite1-replicateA Abundance
- ☑ Sequence

Data

Show abundance values as

- ● Raw
- ○ Relative

Aggregate feature: Name

☐ Hide incomplete features

Aggregate sample: Name

Filter to Selection...
Filter
Select All
Deselect All
Create Abundance Subtable
Create Normalized Abundance Subtable
Create Sequence Sublist
Extract Reads from Selection
Help
Save View...

Figure 4.16: *The ASV abundance table for a single sample*

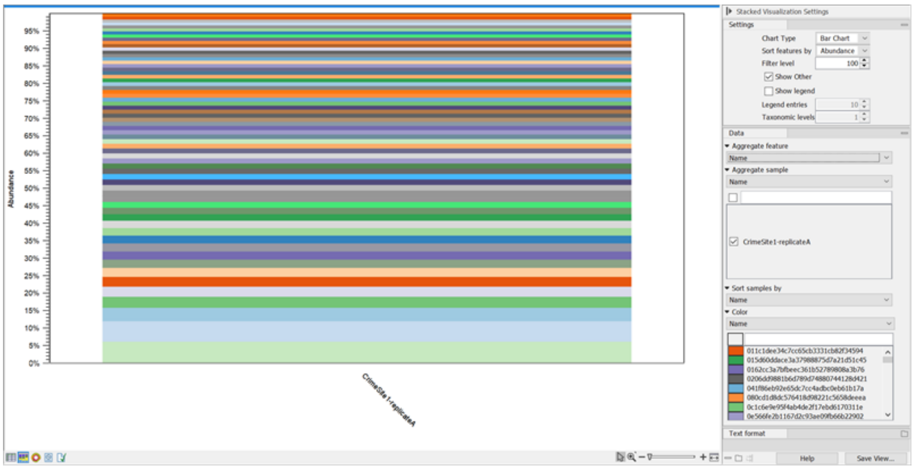
  Below the table, the following actions are available:

  - **Create Abundance Subtable** will create a table containing only the selected rows.
  - **Create Sequence Sublist** will create a sequence list containing only the selected rows.

  For **merged ASV abundance tables**, an additional action is available:

  - **Create Normalized Abundance Subtable** will create a table with all rows normalized on the values of a single selected row. The row used for normalization will disappear from the new abundance table. The normalization scales the abundance table linearly. The scaling factor is calculated by determining the average abundance across all samples and for each sample scale it to the average for the reference. Note that to be enabled, the selected row must have abundance values for all samples. If you have empty values for some samples for the ASVs you wish to use as control, you will need to generate a new abundance table where those samples are not included. If the abundance table is obtained from merging single-sample abundance tables, then the merge should be redone excluding the samples with zero control read counts.

- **Stacked Visualization view** (▦)



Stacked Visualization Settings

Settings

Chart Type: Bar Chart
Sort features by: Abundance
Filter level: 100
☑ Show Other
☐ Show legend
Legend entries: 10
Taxonomic levels: 1

Data

Aggregate feature: Name
Aggregate sample: Name
☑ CrimeSite1-replicateA
Sort samples by: Name
Color: Name

🟧 011c1dee34c7cc65cb3331cb82f34594
🟩 015d60ddace3a37988875d7a21d51c45
🟪 0162cc3a7bfbeec361b52789808a3b76
⬛ 0206dd9881b6d789d74880744128d421
🟦 041f86eb92e65dc7cc4adbc0eb61b17a
🟧 080cd1d8dc576418d98221c5658deeea
🟩 0c1c6e9e95f4ab4de2f17ebd6170311e
🟪 0e566fe2b1167d2c93ae09fb66b22902

Text format

Figure 4.17: *The ASV abundance table Stacked Bar Chart with ASVs aggregated on Name shows the relative abundance of ASVs.*

Adjust the **Side panel** setting **Aggregate feature** to *Name* (figure 4.17) for a visual representation of the relative abundance of ASVs in your sample. Hover over a color to see the name and count of the corresponding ASV.

Use **Filter level** to adjust the number of features shown in the plot. Setting the value to 10 gives you the 10 most abundant ASVs with remaining ASVs grouped as "Other".

For **merged ASV abundance tables**, additional **Side panel** settings may be of use:

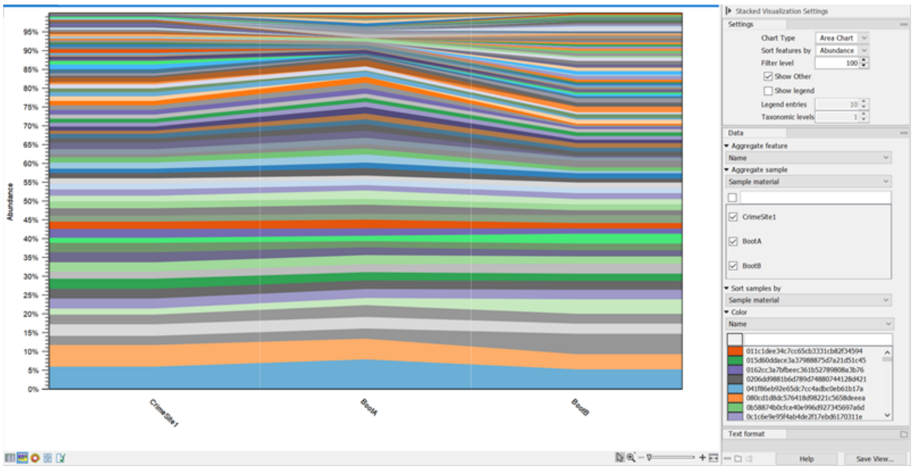
  - With **Chart type** you can switch between *Bar Chart* (figure 4.19) and *Area Chart* (figure 4.18).
  - When **sample metadata** is applied, use **Aggregate sample** to aggregate based on metadata attributes e.g. replicates from the same sample origin.

For **ASV Abundance tables with assigned taxonomies**, you can aggregate ASVs by taxonomy level (figure 4.19).

- **Sunburst view** (◎)

The Sunburst view is only available for **ASV abundance tables with assigned taxonomies**.

The plot is zoomable. Use **Side panel** settings to select how many taxonomy levels to display, and how these should be colored. Lower taxonomy levels will inherit the color from higher levels with different shades. Hover over the plot to view a legend with taxonomy and relative abundances for the highlighted section (figure 4.20).



Stacked Visualization Settings

Settings

Chart Type: Area Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

☐ Show legend

Legend entries: 10

Taxonomic levels: 1

Data

Aggregate feature: Name

Aggregate sample: Sample material

☑ CrimeSite1

☑ BootA

☑ BootB

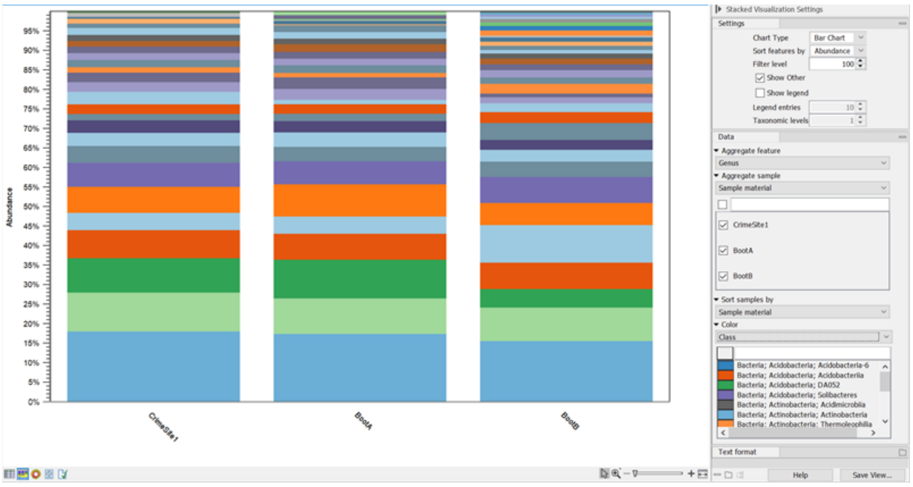
Sort samples by: Sample material

Color: Name

🟥 011c1dee34c7cc65cb3331cb82f34594
🟩 015d60ddace3a37988875d7a21d51c45
🟪 0162cc3a7bfbeec361b52789808a3b76
⬛ 0206dd9881b6d789d74880744128d421
🟦 041f86eb92e65dc7cc4adbc0eb61b17a
🟧 080cd1d8dc576418d98221c5658deeea
🟩 0b58874b0cfce40e996d927345697a6d
🟪 0c1c6e9e95f4ab4de2f17ebd6170311e

Text format

Figure 4.18: *Stacked Area Chart of a merged ASV abundance table Area chart, ASVs aggregated on Name, samples aggregated on Sample material. This shows the ASV abundance at 3 different sample sites for a total of 6 samples.*



Stacked Visualization Settings

Settings

Chart Type: Bar Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

☐ Show legend

Legend entries: 10

Taxonomic levels: 1

Data

Aggregate feature: Genus

Aggregate sample: Sample material

☑ CrimeSite1

☑ BootA

☑ BootB

Sort samples by: Sample material

Color: Class

🟦 Bacteria; Acidobacteria; Acidobacteria-6
🟥 Bacteria; Acidobacteria; Acidobacteriia
🟩 Bacteria; Acidobacteria; DA052
🟪 Bacteria; Acidobacteria; Solibacteres
⬛ Bacteria; Actinobacteria; Acidimicrobiia
🟦 Bacteria; Actinobacteria; Actinobacteria
🟧 Bacteria; Actinobacteria; Thermoleophilia

Text format

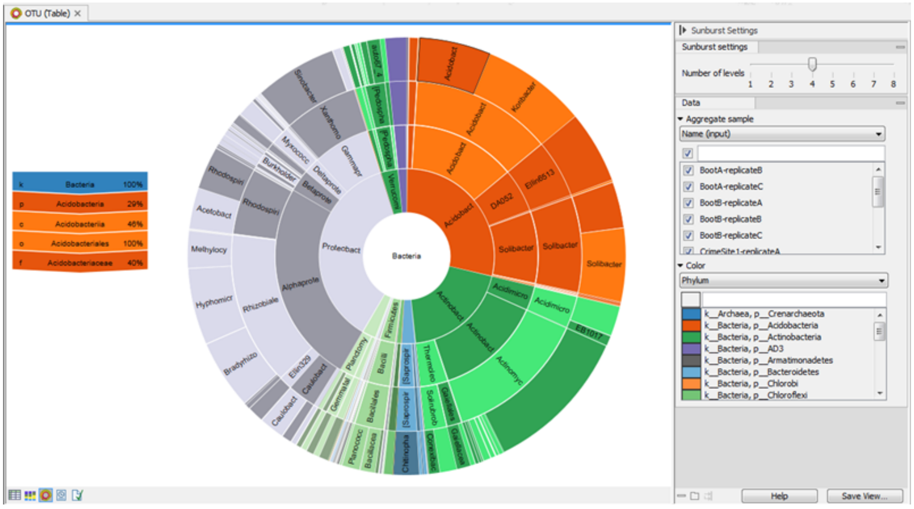
Figure 4.19: *Stacked Bar Chart of a merged ASV abundance tables with assigned taxonomies aggregated on Genus level.*

Click on a lower level field to render that field the center of the plot and display lower level counts in a radial view. Click on the center field to render the level above the current view the center of the view.

4.6.3 Importing ASV abundance tables

You can import files containing amplicon sequence variant counts in tabular format (.xls/.xlsx/.csv) as abundance tables using Standard import:

1. Click on the **Import** (📥) icon in the Toolbar and choose **Standard Import**, or go to **File** | **Import** (📥) | **Standard Import** (📥).
2. Select files to import by clicking on the **Add files** button. Alternatively, specify folders containing files to import by clicking on the **Add folders** button.
3. Choose **Force import as type** and select *ASV abundance table (.xls/.xlsx/.csv)*.



Sunburst Settings

Sunburst settings

Number of levels: 1 2 3 4 5 6 7 8

Data

Aggregate sample: Name (input)

☑

☑ BootA-replicateB
☑ BootA-replicateC
☑ BootB-replicateA
☑ BootB-replicateB
☑ BootB-replicateC
☑ CrimeSite1-replicateA

Color: Phylum

k__Archaea, p__Crenarchaeota
k__Bacteria, p__Acidobacteria
k__Bacteria, p__Actinobacteria
k__Bacteria, p__AD3
k__Bacteria, p__Armatimonadetes
k__Bacteria, p__Bacteroidetes
k__Bacteria, p__Chlorobi
k__Bacteria, p__Chloroflexi

Figure 4.20: *Sunburst view of the microbial community showing all taxa belonging to the kingdom bacteria.*

4. Click on **Finish**.

4.7 Classify Long Read Amplicons

Classify Long Read Amplicons is meant for classification of long-read single-end amplicon sequencing data and was inspired by [Curry et al., 2022]. Reads are mapped to an amplicon reference database of choice and subsequently assigned the most likely taxonomy based on the probability calculated from a series of expectation maximization rounds. The tool can be used for both error-prone and high-accuracy reads.

Note: the tool has not been tested with error-prone PacBio (PacBio CLR) reads, but there is no reason to suspect these reads to be incompatible with the tool. Should you experience issues, please contact QIAGEN Bioinformatics Support team at ts-bioinformatics@qiagen.com.

For tool memory requirements, see (section 1.3). Comprehensive reference databases are needed for reliable species-level resolution, but large databases also require more memory. The tool does not deduplicate the provided reference database, so make sure that databases are non-redundant in order to save memory and runtime.

The underlying algorithm works in five overall steps:

1. **Mapping reads to references**. Reads are mapped to the provided reference database. At this stage each read gets assigned one primary alignment based on mapping parameters, but up to 50 secondary alignments are retained for the following expectation maximization step.

2. **Calculate error model**. An error model of the probability of each alignment type is calculated from all primary alignments. The alignment types are mismatch, insertion, deletion, and softclip.

3. **Initialize alignment probabilities**. For each read-to-taxonomy pair the probability of the alignment is calculated based on the error model. Depending on the composition of the reference database given, a read may map to multiple references with the same taxonomy

i.e., multiple identical read-to-taxonomy pairs exist. In that case, the highest alignment probability between them is used.
In this step the initial probability of the taxonomies is also set with the assumption that all taxonomies in the reference are equally likely.

4. **Expectation maximization**. The algorithm loops through multiple rounds of expectation maximization. In each round, the probability that each read came from that taxonomy for each read-to-taxonomy pair is calculated using the alignment and taxonomy probabilities. The taxonomy probabilities are then updated and the log-likelihood of the estimate is calculated. This loop continues until the increase in log-likeihood falls below the threshold (>0.01) compared to the previous iteration.

5. **Abundance threshold cut-off and reassignment**. When the log-likelihood increase falls below the threshold, a final round of expectation maximization is entered. Here, the taxonomies falling below the set minimum abundance threshold are removed and their assigned abundance is reassigned to the most likely taxonomies among the retained taxonomies.

4.7.1 Classify Long Read Amplicons parameters

To run the tool, go to:

Toolbox | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Amplicon-Based Analysis** ( ) | **Classify Long Read Amplicons** ( )

Classify Long Read Amplicons takes single-end long-read amplicon reads as input. The data should be trimmed for adapters, barcodes, and preferably also primers. Quality scores are not needed for the tool to work and you do not need to quality trim the reads. Since the algorithm infers an error model based on the read alignments, samples from different runs should not be analyzed simultaneously. Instead, you can run samples in "Batch" mode.

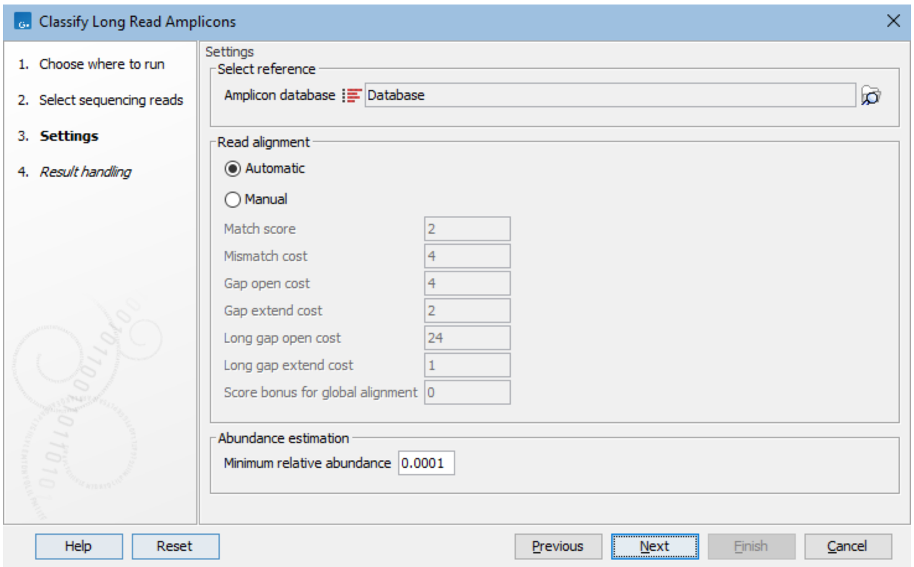
In the wizard, three categories of parameter settings need to be set (figure 4.21):

- **Select reference**. Specify the reference database to be used. Reference databases can be downloaded with the Download Amplicon-Based Reference Database tool (section 14.1) or created by adding taxonomies to sequence lists using the Update Sequence Attributes in Lists tool (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Update_Sequence_Attributes_in_Lists.html).
Reads receive the taxonomy available on the reference sequence assigned i.e., if the reference has any missing taxonomy levels, these will also be missing from the final abundance table.

- **Read alignment**. Select whether to let the mapping algorithm set mapping parameters automatically based on the read platform used, or whether to manually override them. If choosing "Manual", the parameters can be set in the boxes below.

- **Abundance estimation**. Set the "Minimum relative abundance". Any taxa that have relative abundance below this threshold after the expectation maximization loop converges, will be removed and the abundance from these taxa will be reassigned to other probable taxa among the retained taxa.
Note: If you are going to perform differential abundance analysis (section 6.6) following the

classification of amplicons it may be an idea to set "Minimum abundance threshold" to 0. Differential abundance analysis on few samples with few features (taxa) can cause poor dispersion estimates. Should you later want to remove low abundance taxa, you can create an abundance subtable after filtering the table manually using the advanced filters in the top right of the table view.



Settings

Select reference

Amplicon database: Database

Read alignment

● Automatic

○ Manual

Match score: 2

Mismatch cost: 4

Gap open cost: 4

Gap extend cost: 2

Long gap open cost: 24

Long gap extend cost: 1

Score bonus for global alignment: 0

Abundance estimation

Minimum relative abundance: 0.0001

Figure 4.21: *Classify Long Read Amplicons parameter settings.*

4.7.2 Classify Long Read Amplicons output

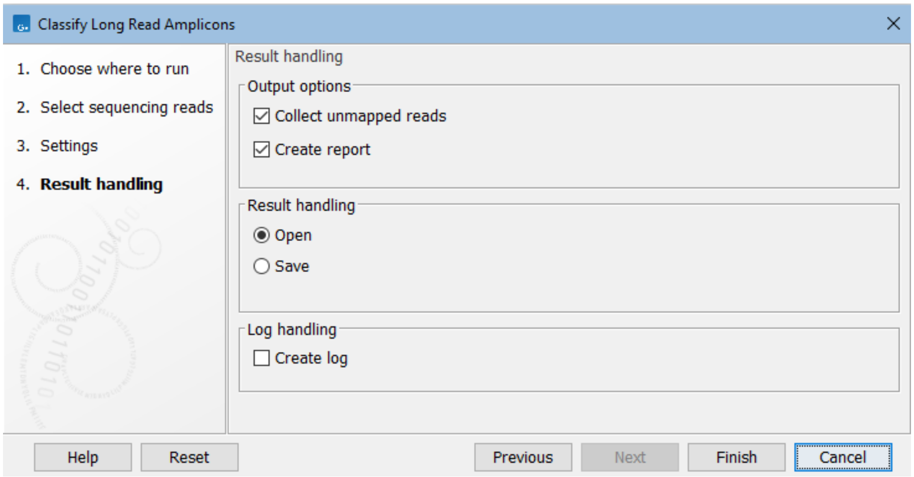
The tool outputs an abundance table. In addition, the following outputs are available in the final wizard step (figure 4.22):

- **Collect unmapped reads.** Outputs a sequence list with all the reads that could not be mapped to the reference database.
- **Create report.** Outputs a summary report.

The Classify Long Read Amplicons abundance table

The abundance table output by **Classify Long Read Amplicons** contains a list of the identified taxonomies that passed the minimum coverage threshold, and the abundance assigned to each taxonomy. Given the probabilistic nature of the algorithm, the reported abundance is not equivalent to a read count, but rather it represents an estimated abundance. The estimate can contain fractions of reads, but the final reported abundance is rounded to the nearest integer.

In contrast to OTU or ASV abundance tables, Classify Long Read Amplicons abundance tables do not contain a sequence for the taxonomies. This is because the reads are not assigned to a sequence but to a taxonomy. Multiple reference sequences may have the same taxonomy (depending on the provided reference database), so all reads assigned to any of the sequences will count towards the abundance for the taxonomy.



Result handling

Output options

☑ Collect unmapped reads

☑ Create report

Result handling

● Open

○ Save

Log handling

□ Create log

Figure 4.22: *Classify Long Read Amplicons output options.*

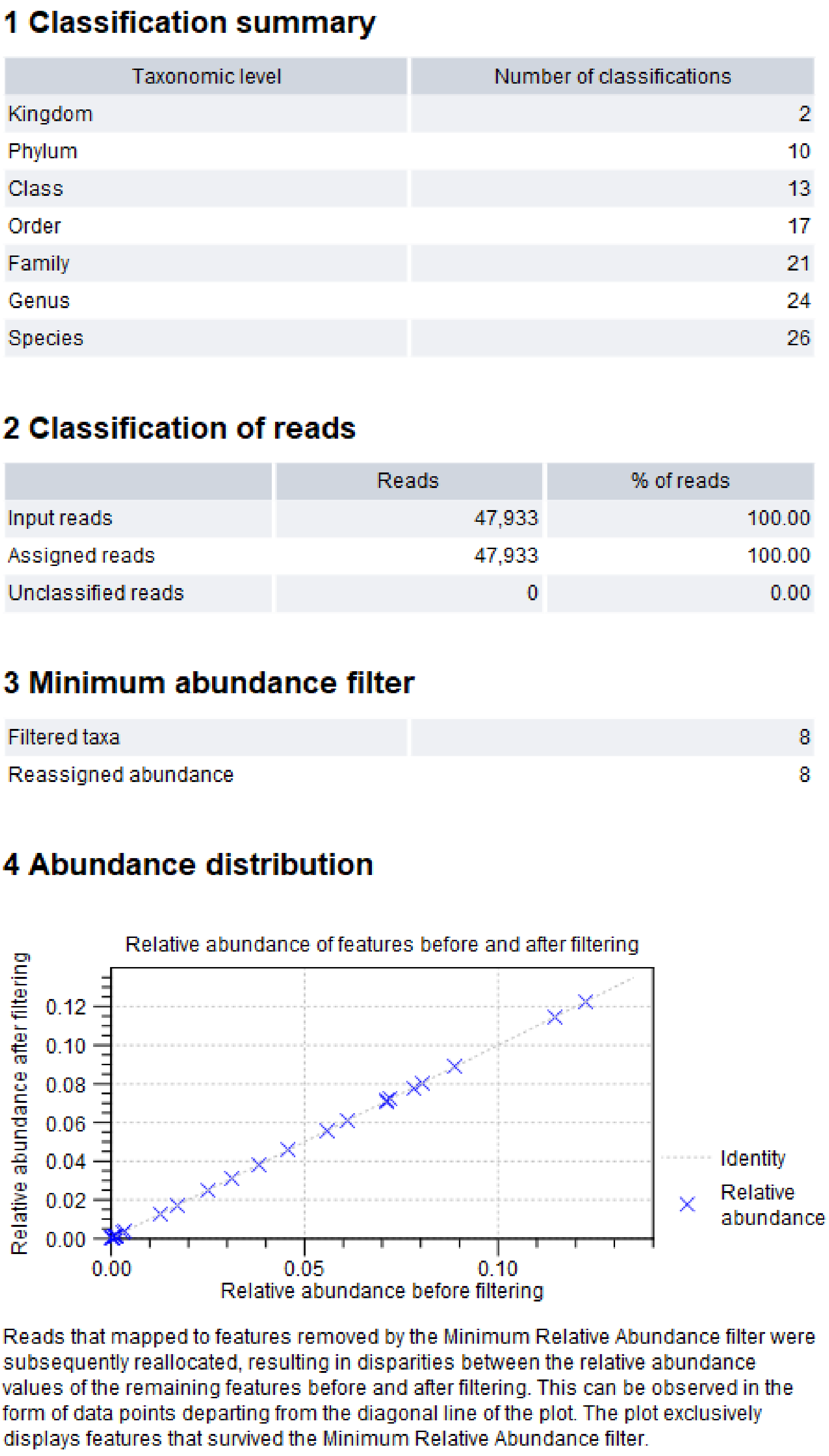
Otherwise, the abundance table contains the same columns, views, and options as the OTU abundance table. See section 4.3.2 for a detailed description of the abundance table options.

The Classify Long Read Amplicons report

An example report can be seen in figure 4.23.

The report contains summary statistics of the results, which can be used for quality checking and verification. It is divided into four sections:

- **Classification summary**. Number of unique taxonomic level reported for each taxonomic level.
- **Classification of reads**. Statistics of the classification given in number of reads and in percentage of input reads.
  - **Input reads**. The number of reads in the input sequence list(s).
  - **Assigned reads**. The number of reads that could be mapped to the reference database.
  - **Unclassified reads**. The number of reads that could *not* be mapped to the reference database. If a large percentage of the reads are unclassified, it could mean that the sample is contaminated, or that the reference database is not comprehensive enough.
- **Minimum abundance filter**.
  - **Filtered taxa**. The number of taxa removed from the final result due to having relative abundance below the minimum abundance threshold.
  - **Reassigned abundance**. The sum of the abundance of the removed taxa which has been reassigned to the most likely taxa among the retained taxa.
- **Abundance distribution**. A scatter plot of the relative abundance of reported features before and after reassigning abundance of features below the minimum abundance threshold. Points will fall on the line or above it, but if one or a few point(s) lie significantly higher than the line, it could mean that that feature has been artificially inflated in the abundance table.



1 Classification summary

Taxonomic level	Number of classifications
Kingdom	2
Phylum	10
Class	13
Order	17
Family	21
Genus	24
Species	26

2 Classification of reads

	Reads	% of reads
Input reads	47,933	100.00
Assigned reads	47,933	100.00
Unclassified reads	0	0.00

3 Minimum abundance filter

Filtered taxa	8
Reassigned abundance	8

4 Abundance distribution

Relative abundance of features before and after filtering
Relative abundance after filtering
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0.00
0.05
0.10
Relative abundance before filtering
Identity
Relative abundance

Reads that mapped to features removed by the Minimum Relative Abundance filter were subsequently reallocated, resulting in disparities between the relative abundance values of the remaining features before and after filtering. This can be observed in the form of data points departing from the diagonal line of the plot. The plot exclusively displays features that survived the Minimum Relative Abundance filter.

Figure 4.23: *Classify Long Read Amplicons report.*

Chapter 5

Taxonomic Analysis

Template workflows for taxonomic analysis are available at:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis ( )**

For more information, see section 2.1.

5.1 Contig Binning

In order to characterize microbial communities, it is key to resolve their composition, diversity and function. With recent advancements in sequencing techniques, whole metagenome shotgun sequencing is becoming standard in metagenomics. Because the output of this technique is a mixture of short DNA fragments belonging to various genomes, computational algorithms for clustering of related sequences are necessary. This approach is globally referred to as sequence binning, and it facilitates downstream analysis steps including: retrieval of metabolic and marker genes; core genome and housekeeping genes analysis; MLST, MLSA and phylogenetic analysis; rRNA and probe design; metagenome re-assembly.

There are two types of binning methods: a) taxonomy dependent and b) taxonomy independent. The first is implemented here through the Bin Pangenomes by Taxonomy tool and the second via the Bin Pangenomes by Sequence tool [Sedlar et al., 2017]. The performance of approach a) is limited to the completeness of an existing database, whereas approach b) usually suffers from a lack of precision. In order to leverage the full strength of the two approaches a combined analysis is encouraged. The template workflow **QC, Assemble and Bin Pangenomes** (section 2.1.4) facilitates this as it employs both methodologies to generate lists of contigs of assembled, binned contigs.

5.1.1 Bin Pangenomes by Taxonomy

This tool assigns contigs and the reads they are composed of into bins with other contigs presumably of closely related taxonomy. For this we use a microbial reference (genome) sequence database, which comprises sequences with taxonomic information. Furthermore, in order to separate contigs that originate from plasmids from those of genomic origin, the Bin Pangenomes by Taxonomy tool additionally takes a plasmid database as input.

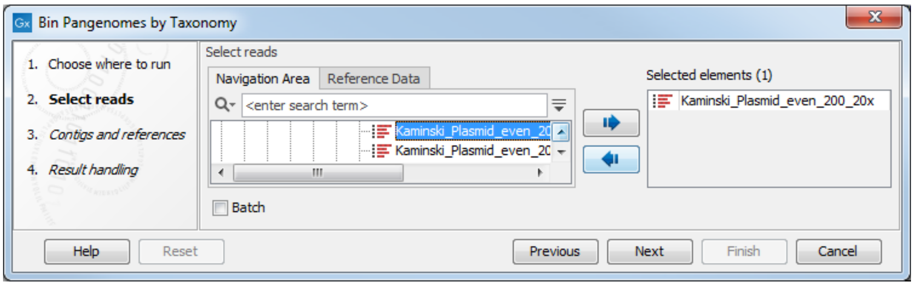
Binning occurs in 5 consecutive steps:

1. Obtain taxonomic information for reads
2. Obtain plasmid information for reads
3. Map reads to contigs
4. Assign taxonomic and plasmid labels to contigs
5. Group and filter contigs according to labels (Contig purity)

To start the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Bin Pangenomes by Taxonomy** ( )

The Bin Pangenomes by Taxonomy takes one or several single or paired-end read files as input (figure 5.1).



Bin Pangenomes by Taxonomy

1. Choose where to run
2. **Select reads**
3. *Contigs and references*
4. *Result handling*

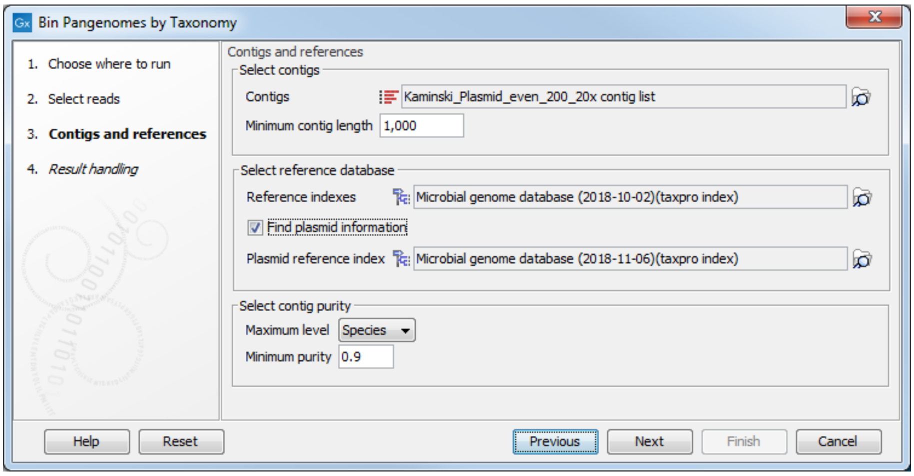
Select reads

Selected elements (1): Kaminski_Plasmid_even_200_20x

□ Batch

Figure 5.1: *Select the reads.*

The tool is designed to work on contigs assembled from the same set of reads used as input, previously assembled using the De Novo Assembly Metagenome tool (as in the workflow, see section 2.1.4). You can also specify here the minimum contig length desired (figure 5.2).



Bin Pangenomes by Taxonomy

1. Choose where to run
2. Select reads
3. **Contigs and references**
4. *Result handling*

Contigs and references

Select contigs
Contigs: Kaminski_Plasmid_even_200_20x contig list
Minimum contig length: 1,000

Select reference database
Reference indexes: Microbial genome database (2018-10-02)(taxpro index)
☑ Find plasmid information
Plasmid reference index: Microbial genome database (2018-11-06)(taxpro index)

Select contig purity
Maximum level: Species
Minimum purity: 0.9

Figure 5.2: *Select the references and specify the parameters needed for running the tool.*

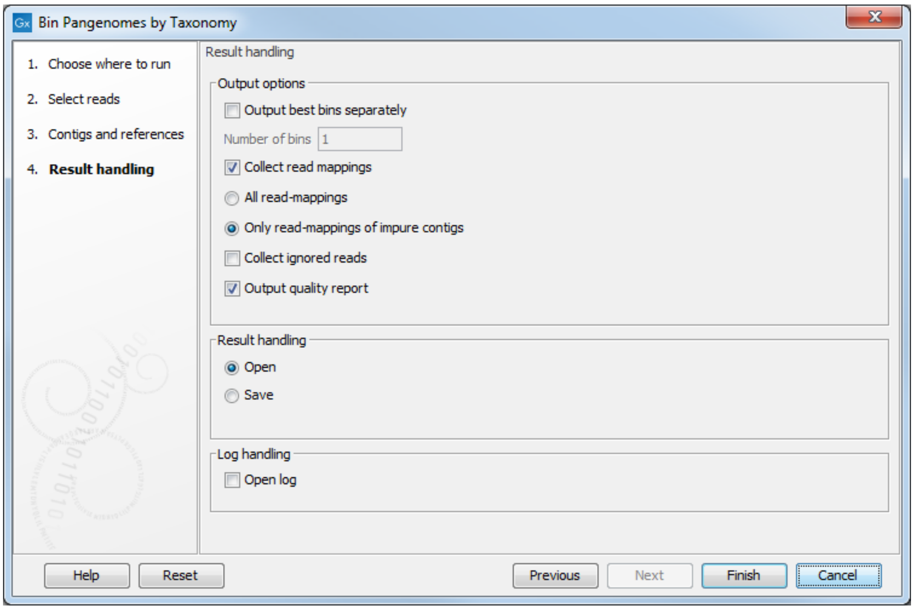
As reference databases, one or two Taxonomic Profiling index files can be provided:

- the file provided as "Reference indexes" is used to find taxonomic information for the reads
- the file provided as "Plasmid reference index" (once the "Find plasmid information option is checked) is used to distinguish genomic reads from plasmid reads.

Both references can be obtained by using the Download Curated Microbial Reference Database tool (section 15.1) or Download Custom Microbial Reference Database tool (section 15.2). If using the latter, the indexes can be built with the Create Taxonomic Profiling Index tool (section 15.4).

Depending on the dataset, it may be necessary to adapt the contig purity settings, where "Maximum level" refers to a maximum level in the taxonomic tree and where a specific "Minimum purity" per contig needs to be reached in order for it to be considered a part of a bin. For example, if Maximum level = Genus and Minimum purity = 0.8 and 512 reads map to a given contig, at least 0.8 * 512 = 410 reads need to have the same Genus level taxonomy in order for the contig to become part of the respective bin. If more precise taxonomic information is available (e.g., on Species level) with the requested minimum purity, this information will be used instead.

The "Result handling" dialog allows you to specify outputs (figure 5.3):



Result handling

Output options

□ Output best bins separately

Number of bins 1

☑ Collect read mappings

○ All read-mappings

● Only read-mappings of impure contigs

□ Collect ignored reads

☑ Output quality report

Result handling

● Open

○ Save

Log handling

□ Open log

Figure 5.3: *Specify the outputs needed.*

- Choose to output a certain number of the best bins separately, which means that a chosen number of bins will be written to separate outputs. In this context, "best" is defined by completeness, estimated as the number of contig nucleotides in bin divided by the number of nucleotides in the assigned reference genome.
- Specify whether to collect the read mappings and which kind (all, only impure contigs)
- Collect ignored reads (i.e., reads not mapping to contigs)
- Output a quality report for the bins, where bins are ordered in order of completeness (see above).

The standard output of the Bin Pangenomes by Taxonomy tool consists of a list of (binned) contigs and one sequence list per input reads file (or two for paired reads) where each of the sequences is labeled according to its most probable origin and bin it ended up in (the bin annotation is stored as "Assembly ID" annotation in order for it to work seamlessly with other tools). Also, a column called "isPlasmid" provides a true/false label whether the contig/read was mapped respectively to a plasmid or a genome. The tool can also output a Taxonomy binning report.

5.1.2 Bin Pangenomes by Sequence

Binning by sequence is done irrespective of a database, only depending on content and coverage. To have both sources of information available, the Bin Pangenomes by Sequence tool takes read mappings to contigs as input, where there should be one read mapping per technical replicate (each mapping to the same contigs) in order to make most use of coverage information across all samples. However, if read mappings are not available, the Bin Pangenomes by Sequence tool also takes plain sequence lists of contigs as input.

The Bin Pangenomes by Sequence algorithm is based on the MetaBAT [Kang et al., 2015] and SCIMM [Kelley and Salzberg, 2010] algorithms with several modifications:

- A logistic regression model (simliar to MetaBAT) may use a variable number of parameters. The number of trusted parameters is adjusted to the number of contigs in a bin and the parameters are adjusted during the algorithm. Only Kmer features are used.
- The interpolated Markov Models of SCIMM are replaced by variable order markov models.
- Random Projections are used to speed up the search for the centers in the proximity of a contig in combined Kmer-Coverage space.
- Poisson-Mixture models are used to fit the coverage distribution on contigs.

The tool was designed for sample sizes in the order of five million reads mapped to 1000 contigs. It does not support substantially bigger data sets.

To start the tool, go to:

Toolbox | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Bin Pangenomes by Sequence** ( )

The Bin Pangenomes by Sequence takes one sequence list of contigs or one read mapping per sample as input (figure 5.4).

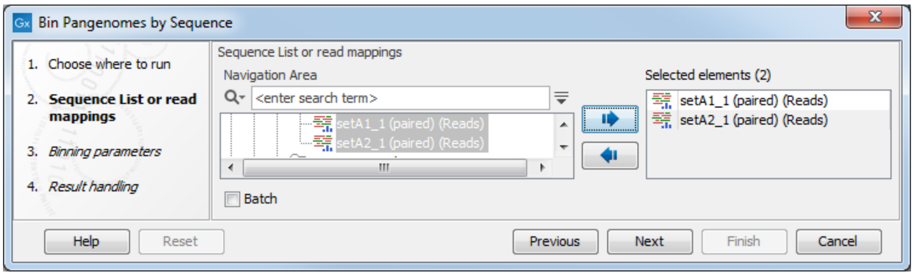
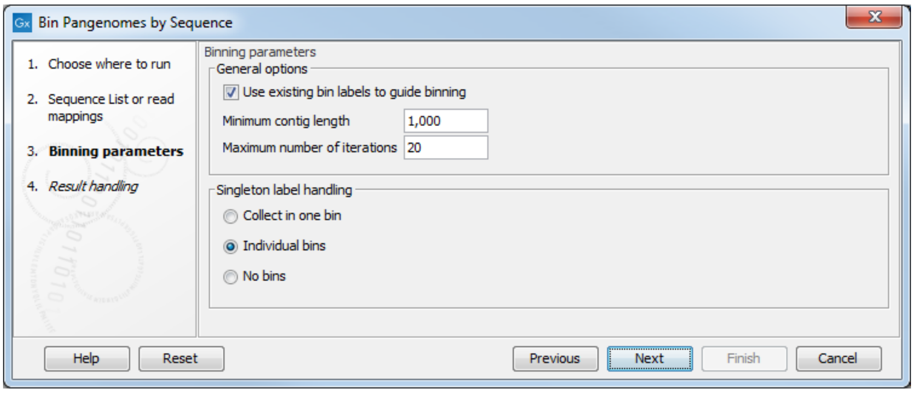


Figure 5.4: *Select the contigs or read mappings.*

In the next dialog (figure 5.5), the several parameters can be specified:



Binning parameters

General options

☑ Use existing bin labels to guide binning

Minimum contig length 1,000

Maximum number of iterations 20

Singleton label handling

○ Collect in one bin

● Individual bins

○ No bins

Figure 5.5: *Configuration of the Bin Pangenomes by Sequence.*

- **Use existing labels to guide binning** may be used to improve binning quality and speed. For read mapping inputs, labels assigned to the reads by the Bin Pangenomes by Taxonomy are used, while for sequence list inputs the Assembly ID (see section 21) labels assigned to the contigs are used.
- **Minimum contig length** specifies the minimal length for contigs to be considered (should be at least 1000 to obtain decent bin qualities
- **Maximum number of iterations** specifies how many purification steps at most should be made.
- **Singleton label handling** decides whether singletons should be collected in one bin, kept in individuals bin, or not included in any bins.

Finally, in the "Result handling" dialog, it may be specified whether the reads of the binned contigs should be labelled and collected.

The standard output of the Bin Pangenomes by Sequence tool consists of a Sequence binning report, a contig list with their assigned bin listed in the Assembly_ID column, and as many read lists as read mappings were used as input in the tool, where reads have been assigned the bin of the contig they belong to.

5.2 Taxonomic Profiling

Taxonomic profiling provides insight into the taxonomic composition of whole metagenome samples and estimates the relative abundance of the detected taxa.

Reads are mapped to a reference genome database and are assigned to a reference genome or higher taxonomy level based on their mapping quality score, i.e. the confidence that the read is correctly mapped:

- Reads that map to only one reference location are assigned to that genome.
- Reads that map best to one reference location, but where other almost as good alternatives are found, are assigned to the taxonomy one level up from the best-match genome.

- Reads that map equally well to more than one reference location are assigned to the lowest common ancestor.

If a host genome is provided, reads that map better to this are filtered. Reads are mapped individually to the reference genome database and the host genome. Reads that map to both are assigned to the match with higher mapping score.

For paired reads, when a read pair is broken, either because only one read in the pair matches, or the distance or relative orientation is wrong, both reads are discarded.

Following mapping of reads, qualification and quantification steps refine the results:

- **Qualification**. Determines whether a particular taxon is represented in a sample. This calculation is based on a confidence scores; of whether a reference sequence was assigned reads by pure chance. Any taxon with a confidence score < 0.995 will be ignored and reads will be reassigned to its closest qualified ancestor. By construction, the confidence score is very close to 1.0 except on the Kingdom level of the taxonomy, thus it is not reported.
- **Quantification**. Calculates the abundance of qualified taxa based on the number of assigned reads.

  For data sets with varying read length, the abundance values may optionally be adjusted to correct for a skewed read distribution between taxa, see *Adjust for read length variation* in section 5.2.1.

To run the Taxonomic Profiling tool, go to

Toolbox | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Taxonomic Profiling** ( )

In the first dialog, select the sequence list to analyze (figure 5.6).

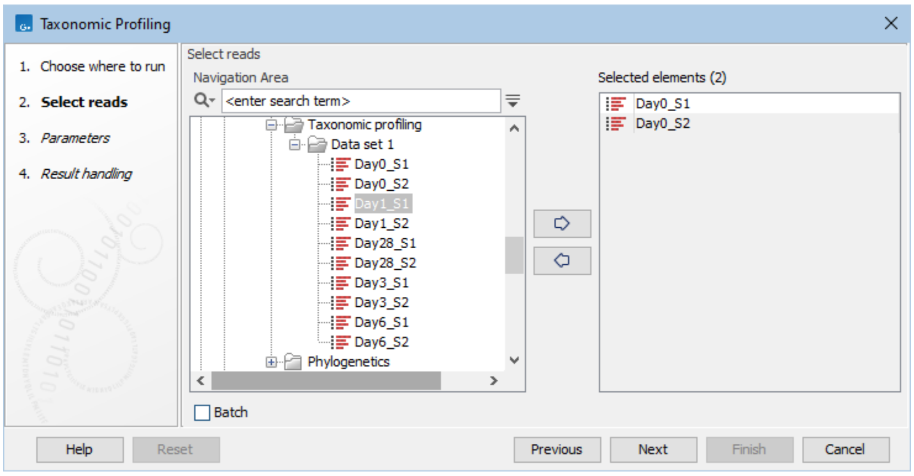


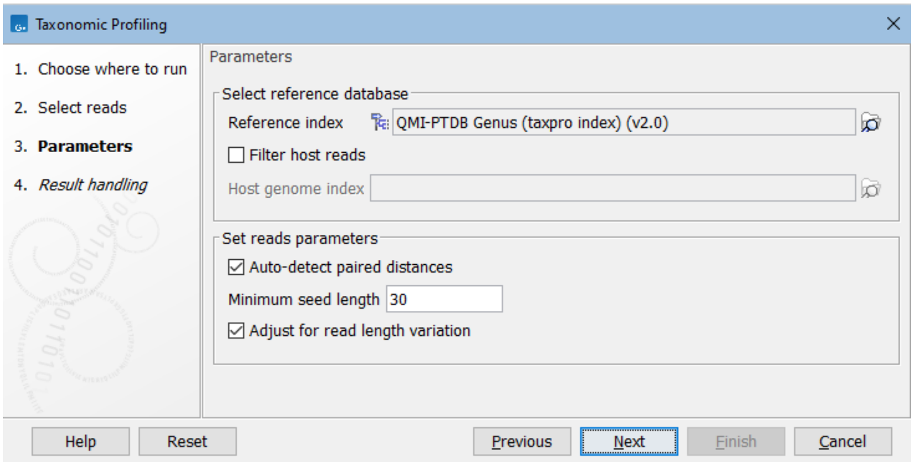
Figure 5.6: *Select a sequence list as input.*

5.2.1 Taxonomic Profiling parameters

When sequences are selected, click **Next**, and you will see the dialog in (figure 5.7).

Select reference databases:

- **Reference index.** The database of reference genomes that you are analyzing for.
- **Filter host reads.** If this is checked, reads that map better to the specified host genome are filtered and will not count toward taxonomic results.
- **Host genome index.** The host genome, or background genome, represents the reference sequences that may be present in your sample, but that are not the target of your analysis. As an example, to analyze the microbiome from human gut samples, you would specify the human reference index to filter reads that map against the human genome.



Taxonomic Profiling

1. Choose where to run
2. Select reads
3. **Parameters**
4. *Result handling*

Parameters

Select reference database

Reference index: QMI-PTDB Genus (taxpro index) (v2.0)

□ Filter host reads

Host genome index:

Set reads parameters

☑ Auto-detect paired distances

Minimum seed length: 30

☑ Adjust for read length variation

Figure 5.7: *Set the parameters for taxonomic profiling.*

Curated reference databases and indexes are available with the Download Curated Microbial Reference Database tool (section 15.1). Alternatively, you can use the Download Custom Microbial Reference Database tool (section 15.2) to create your own custom reference database. To create indexes from reference databases and host genomes, use the Create Taxonomic Profiling Index tool (section 15.4).

Under *Set reads parameters*, the following options are available:

- **Auto-detect paired distances.** For paired data, this default choice will automatically calculate the distance between reads in pairs as follows:

  1. A sample of 100,000 reads is extracted randomly from the full data set and mapped against the reference index using a very wide distance interval.
  2. The distribution of distances between the paired reads is analyzed using a method that investigates the shape of the distribution and finds the 0.5% boundaries of the peak. These values make up the distance interval. If fewer than 10,000 reads are mapped as pairs, the range is calculated using the standard deviation.
  3. The full sample is mapped using the calculated distance interval.
  4. The history of the result records the distance interval used.

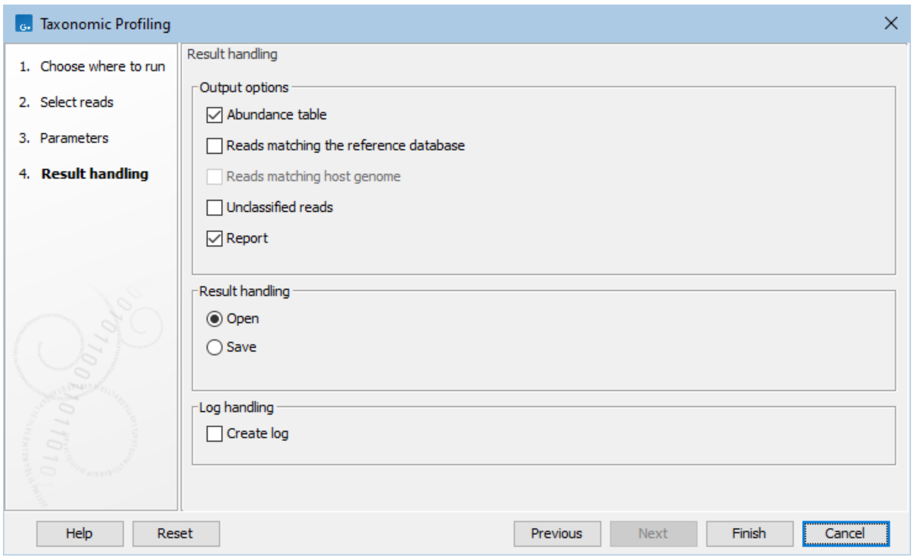
  If the automatic detection of paired distances is not checked, the tool will use the information about minimum and maximum distance recorded on the input sequence lists.

- **Minimum seed length**. The minimum number of nucleotides with which a read must map to a reference sequence for it to be considered a valid match. Increasing this value will give higher precision of called taxa (true positives). Lowering it will result in more taxa being called, but at the cost of precision (more false positives).
  Apart from the Minimum seed length parameters, reads are mapped with standard read mapping parameters (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters.html).

- **Adjust for read length variation**. This option is recommended for data sets with varying read lengths to adjust for skewed read distribution between taxa. If checked, abundance and coverage values are adjusted by weighting the reads assigned to a taxon by the number of nucleotides mapped to the taxon. Calculation of abundance and coverage with and without this option checked is explained in section 5.2.3.

5.2.2 Taxonomic Profiling output

Clicking **Next** will allow you to specify the output as shown in figure 5.8.


Taxonomic Profiling

1. Choose where to run
2. Select reads
3. Parameters
4. **Result handling**

Result handling

Output options
☑ Abundance table
□ Reads matching the reference database
□ Reads matching host genome
□ Unclassified reads
☑ Report

Result handling
● Open
○ Save

Log handling
□ Create log

Figure 5.8: *Specify the output.*

The following outputs are available:

- **Abundance table**. The main result with identified taxa and associated abundance values (*taxonomic profile*).
- **Reads matching the reference database**. Creates a sequence list for each input with reads that were assigned to the reference database (*database matches*).
- **Reads matching host genome**. Creates a sequence list for each input with reads that were assigned to the host genome (*host matches*).
- **Unclassified reads**. Creates a sequence list for each input with unassigned reads (*unclassified reads*).
- **Report**. Creates a summary report.

Sequence list output

The sequence lists with reads that were assigned to the reference database and host genome contain the following annotations:

- **Mapping Quality Score.** Reads with quality score <10 will have been assigned to a higher taxonomy level.
- **Mapping Score.** The score for the read alignment (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters.html).

In addition, the list of reference database reads contains the **Taxonomy** annotation with the full taxonomy of the taxon to which the read was assigned.

Taxonomic Profiling report

The Taxonomic Profiling report (figure 5.9) contains information about the taxonomic profiling run and databases used.

- **Taxonomic summary.** The number of detected taxa for each taxonomic level.
- **Classification of reads.** Information about the number of reads that were assigned to reference and host databases, or left unclassified. For database matches, both the total number of reads and the number of uniquely matched reads are provided.
- **Reference database summary.** Information about the reference database. Reads that map across Kingdoms will count as a reference database match, but will not contribute to values in the abundance table as no meaningful taxon can be assigned.
- **Host database summary.** Information about the host genome.
- **Auto-detect paired distances.** The calculated paired distance range (provided when the corresponding option was applied).

5.2.3 Taxonomic Profiling abundance table

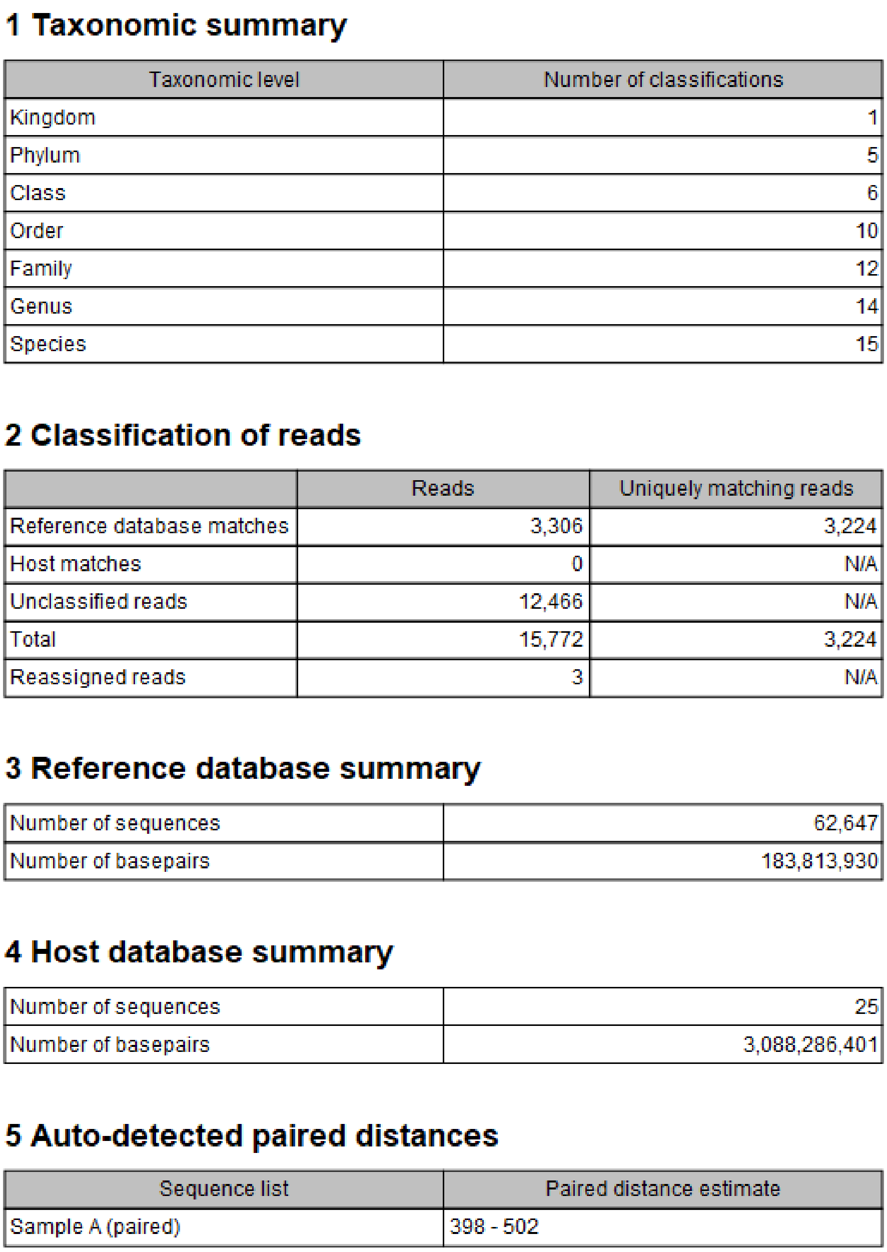
The Taxonomic profiling abundance table lists the taxa identified in your sample. To go beyond single sample view, use the Merge Abundance Tables tool to create a multi-sample abundance table (section 6.1). Some of the options mentioned below are relevant for such multi-sample abundance tables only.

The abundance table can be vizualized in Table view (▦), Stacked Visualization view (▥), and Sunburst view (◎).

Table view (▦)

The table displays a number of columns, some of which are available only when the table is not aggregated by taxonomy (figure 5.10):

- **ID.** The ID of the reference genome or taxon.



1 Taxonomic summary

Taxonomic level	Number of classifications
Kingdom	1
Phylum	5
Class	6
Order	10
Family	12
Genus	14
Species	15

2 Classification of reads

	Reads	Uniquely matching reads
Reference database matches	3,306	3,224
Host matches	0	N/A
Unclassified reads	12,466	N/A
Total	15,772	3,224
Reassigned reads	3	N/A

3 Reference database summary

Number of sequences	62,647
Number of basepairs	183,813,930

4 Host database summary

Number of sequences	25
Number of basepairs	3,088,286,401

5 Auto-detected paired distances

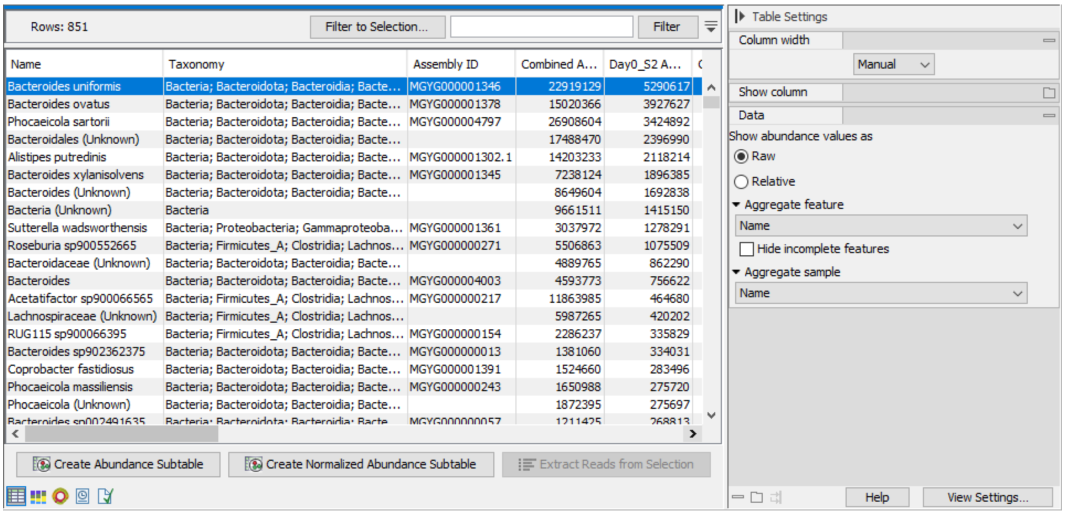
Sequence list	Paired distance estimate
Sample A (paired)	398 - 502

Figure 5.9: *The Taxonomic Profiling report.*

- **Name**. The name of the taxon as specified by the reference database.
  If the name contains the text "(Unknown)", this indicates that the taxon corresponds to a higher-level node in the taxonomy, and that this node had a significant amount of reads associated to ancestor taxa that are present in the database but were disqualified. This indicates that there was some organism in the sample for which there is no exact match in the reference database, but is most likely closely related to this taxon. If the name does not contain the text "(Unknown)", it means that the sample contains this exact taxon.

- **Taxonomy**: The taxonomy of the taxon as specified by the reference database.

- **Assembly ID**: The ID of the assembly as specified by the reference database. Typically a



Rows: 851

Name	Taxonomy	Assembly ID	Combined A...	Day0_S2 A...
Bacteroides uniformis	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000001346	22919129	5290617
Bacteroides ovatus	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000001378	15020366	3927627
Phocaeicola sartorii	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000004797	26908604	3424892
Bacteroidales (Unknown)	Bacteria; Bacteroidota; Bacteroidia; Bacte...		17488470	2396990
Alistipes putredinis	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000001302.1	14203233	2118214
Bacteroides xylanisolvens	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000001345	7238124	1896385
Bacteroides (Unknown)	Bacteria; Bacteroidota; Bacteroidia; Bacte...		8649604	1692838
Bacteria (Unknown)	Bacteria		9661511	1415150
Sutterella wadsworthensis	Bacteria; Proteobacteria; Gammaproteoba...	MGYG000001361	3037972	1278291
Roseburia sp900552665	Bacteria; Firmicutes_A; Clostridia; Lachnos...	MGYG000000271	5506863	1075509
Bacteroidaceae (Unknown)	Bacteria; Bacteroidota; Bacteroidia; Bacte...		4889765	862290
Bacteroides	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000004003	4593773	756622
Acetatifactor sp900066565	Bacteria; Firmicutes_A; Clostridia; Lachnos...	MGYG000000217	11863985	464680
Lachnospiraceae (Unknown)	Bacteria; Firmicutes_A; Clostridia; Lachnos...		5987265	420202
RUG115 sp900066395	Bacteria; Firmicutes_A; Clostridia; Lachnos...	MGYG000000154	2286237	335829
Bacteroides sp902362375	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000000013	1381060	334031
Coprobacter fastidiosus	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000001391	1524660	283496
Phocaeicola massiliensis	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000000243	1650988	275720
Phocaeicola (Unknown)	Bacteria; Bacteroidota; Bacteroidia; Bacte...		1872395	275697
Bacteroides sp002491635	Bacteria; Bacteroidota; Bacteroidia; Bacte...	MGYG000000057	1211425	268813

Table Settings

Column width: Manual

Show column

Data

Show abundance values as

● Raw

○ Relative

Aggregate feature: Name

☐ Hide incomplete features

Aggregate sample: Name

Filter to Selection...
Filter
Create Abundance Subtable
Create Normalized Abundance Subtable
Extract Reads from Selection
Help
View Settings...

Figure 5.10: *Taxonomic profiling abundance table.*

GenBank assembly accession number.

- **Combined Abundance**: Total abundance across samples.
- **Min**, **Max**, **Mean**, **Median** and **Std**. Minimum, maximum, mean, median and standard deviation of abundance values across samples.
- **Abundance**. Number of reads assigned to the taxon as counted during the quantification phase, see section 5.2. If the option *Adjust for read length variation* is checked, the abundance value will be adjusted by weighing the reads assigned to a taxon by the total number of nucleotides mapped to the taxon:

  – Adjusted abundance = (abundance in nucleotides) / (average mapped read length)

  For data sets where all reads have similar length, the adjusted abundance will be very similar to the raw read count. Occasionally, weighting may lead to zero reads assigned to a qualified taxon if there are only few, shorter than average reads assigned to this taxon.

- **Coverage**: The coverage estimate of the sample. Coverage calculation depends on the representation of the taxon:

  – The taxon is represented by a single-sequence genome:

    ∗ Coverage = (Weighted nucleotides matching the genome sequence) / (genome sequence length).
      The weight is adjusted based on the number of ambiguous matches for the individual reads. A unique match equals a maximum weight of 1.

  – The taxon is represented by a multi-sequence genome:

    ∗ *Adjust for read length variation* is checked: Coverage = (total number of nucleotides assigned to the genome sequences) / (total genome sequence length).

    ∗ *Adjust for read length variation* is unchecked: Coverage = ((total number of reads assigned to the genome sequences) x (average sample read length)) / (total genome sequence length).

- The taxon is a parent of filtered, unqualified genomes. The reads from the filtered genomes were reassigned to the parent taxon:
  - *Adjust for read length variation* is checked: Coverage = (total number of nucleotides initially assigned to the filtered genomes) / (average sequence length of filtered genomes).
  - *Adjust for read length variation* is unchecked: Coverage = ((total number of reads initially assigned to the filtered genomes) x (average sample read length)) / (average sequence length of filtered genomes).

The **Side panel** offers the following settings:

- **Show abundance values as**. Switch between Raw and Relative abundance. Relative abundance is calculated as: Relative abundance = (Abundance) / (Sum of abundances).
- **Aggregate feature**. Select a taxonomic level to aggregate abundance values by. For example, select *Family* to display abundance values per Family as opposed to per genome.
- **Hide incomplete features**. Hides features that are not resolved to the taxonomic level selected with the option above.
- **Aggregate sample**. Select a metadata attribute to aggregate samples with same metadata value into one column with combined abundance values.

Below the table, the following actions are available:

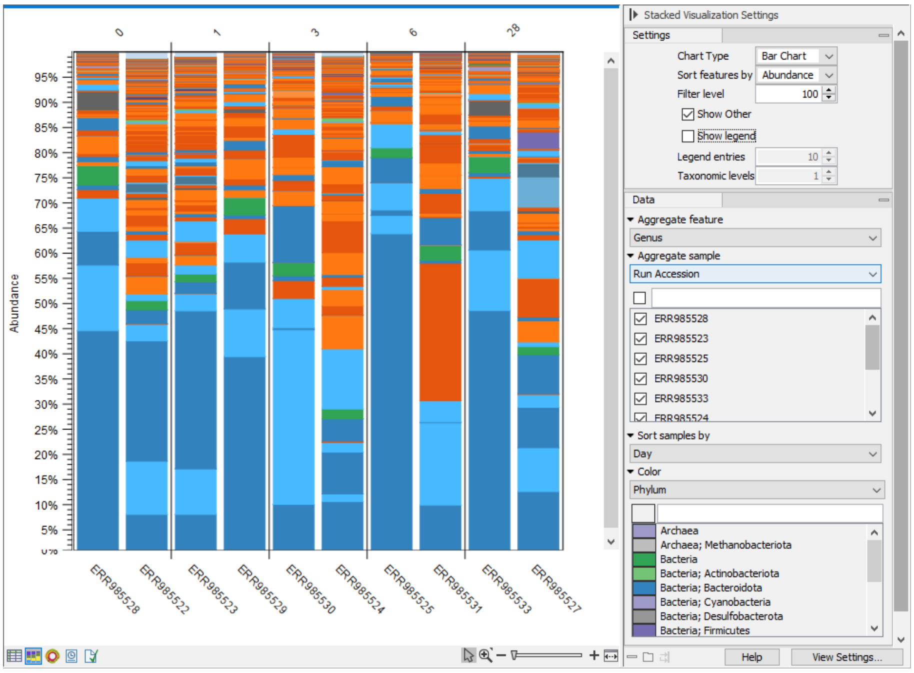
- **Create Abundance Subtable**. Creates a table containing only the selected rows.
- **Create Normalized Abundance Subtable**. Creates a table with all rows normalized by the values of a single selected row. The row used for normalization will disappear from the new abundance table. The normalization scales the abundance table linearly, where the scaling factor is calculated by determining the average abundance across all samples and for each sample scale it to the average for the reference. Note that to be enabled, all abundance values in the selected rows must be larger than zero.
- **Extract Reads from Selection**. Extracts reads that were uniquely associated with the selected rows. This option is available if you selected the output *Reads matching the reference database* in the Taxonomic Profiling tool dialog.

Stacked Visualization view (▦)

The Stacked Vizualization view displays the relative abundance of each feature. Use the **Side panel** setting **Bar type** to switch between Bar Chart (figure 5.11) and Area Chart (figure 5.12). Different colored bars or areas represent different features. A column represents a sample or - if aggregated by sample level - a group of samples.

Hold your pointer over an area to have the full taxonomy and abundance value displayed in a tooltip.

You can adjusted the view further via the **Side panel** settings. Selected options are:



Stacked Visualization Settings

Settings

Chart Type: Bar Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

□ Show legend

Legend entries: 10

Taxonomic levels: 1

Data

Aggregate feature

Genus

Aggregate sample

Run Accession

□

☑ ERR985528

☑ ERR985523

☑ ERR985525

☑ ERR985530

☑ ERR985533

☑ ERR985524

Sort samples by

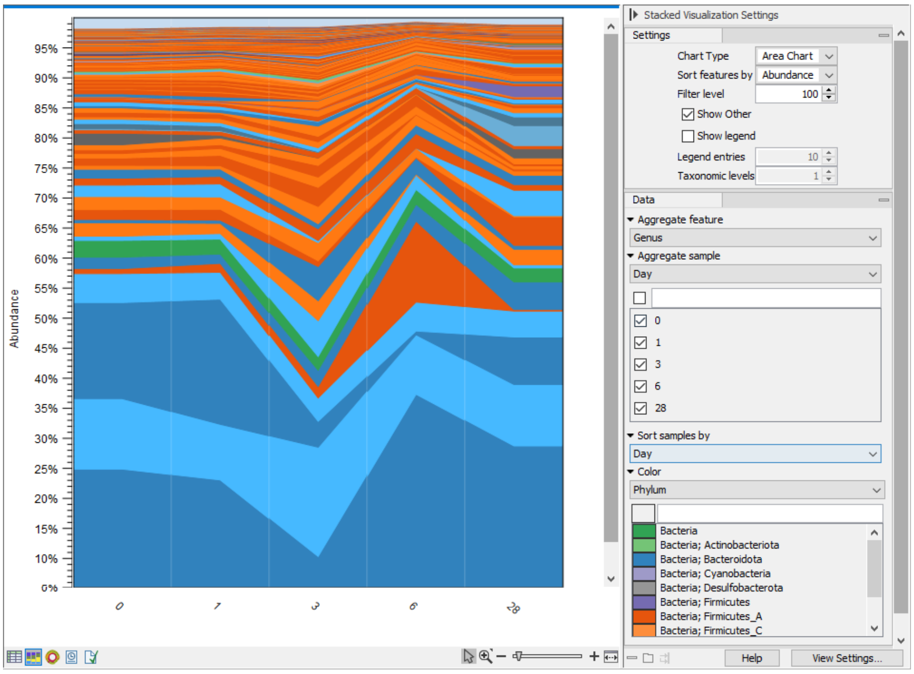
Day

Color

Phylum

Archaea
Archaea; Methanobacteriota
Bacteria
Bacteria; Actinobacteriota
Bacteria; Bacteroidota
Bacteria; Cyanobacteria
Bacteria; Desulfobacterota
Bacteria; Firmicutes

Figure 5.11: *Stacked bar chart.*



Stacked Visualization Settings

Settings

Chart Type: Area Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

□ Show legend

Legend entries: 10

Taxonomic levels: 1

Data

Aggregate feature

Genus

Aggregate sample

Day

□

☑ 0

☑ 1

☑ 3

☑ 6

☑ 28

Sort samples by

Day

Color

Phylum

Bacteria
Bacteria; Actinobacteriota
Bacteria; Bacteroidota
Bacteria; Cyanobacteria
Bacteria; Desulfobacterota
Bacteria; Firmicutes
Bacteria; Firmicutes_A
Bacteria; Firmicutes_C

Figure 5.12: *Stacked area chart.*

- **Aggregate feature**. Select a taxonomic level to aggregate abundance values by. For example, select *Family* to have each section in the plot represent a Family instead of a genome.

- **Aggregate sample**. Select a metadata attribute to aggregate samples with same metadata value into one column with combined abundance values. (Relvant for multi-sample abundance tables only). Use the checkboxes below to specify which samples or groups of samples to include in the plot.

- **Sort samples by**. Select a metadata attribute to sort samples by the corresponding attribute values. The values are listed above the plot (figure 5.11).
- **Color**. Select a taxonomic level to color the plot by. As an example, if you select *Phylum*, all features belonging to the same Phylum will get different shades of the same base color.

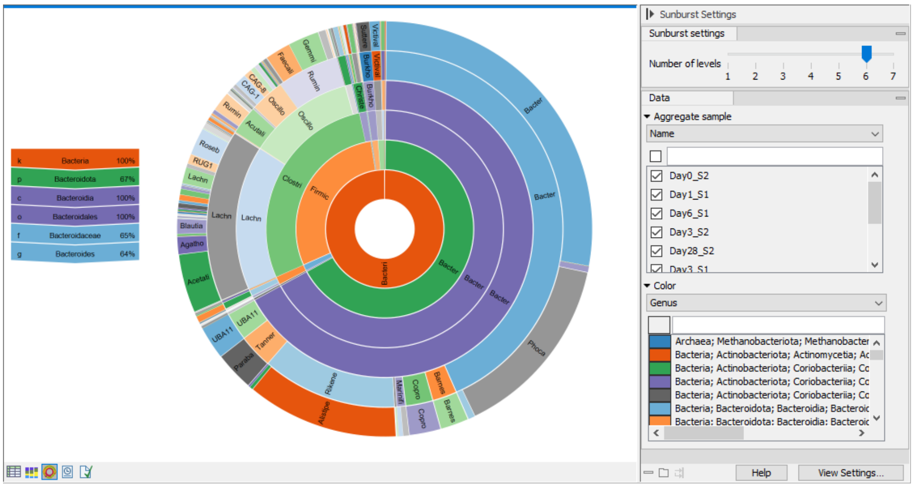
Sunburst view ( )

The plot is zoomable. Click on a section to zoom in and render the plot with this section at the center. Click on the center of the plot to zoom out one level at a time.

Hold your pointer over the plot to have the legend reflect the highlighted section (figure 5.13).

Use the **Side panel** settings to adjust the plot:

- **Number of levels**. Select the maximum number of taxonomic levels to display.
- **Aggregate sample**. Select a metadata attribute to group aggregate samples by, and use the checkboxes below to specify which samples or groups of samples to include in the plot.
- **Color**. Select a taxonomic level to color the plot by. Lower levels will inherent the color and get different shades of the same color.



k	Bacteria	100%
p	Bacteroidota	67%
c	Bacteroidia	100%
o	Bacteroidales	100%
f	Bacteroidaceae	65%
g	Bacteroides	64%

Sunburst Settings

Sunburst settings

Number of levels: 1 2 3 4 5 6 7 (6 selected)

Data

Aggregate sample: Name

☐

☑ Day0_S2
☑ Day1_S1
☑ Day6_S1
☑ Day3_S2
☑ Day28_S2
☑ Day3_S1

Color: Genus

Archaea; Methanobacteriota; Methanobacter
Bacteria; Actinobacteriota; Actinomycetia; Ac
Bacteria; Actinobacteriota; Coriobacteriia; Cc
Bacteria; Actinobacteriota; Coriobacteriia; Cc
Bacteria; Actinobacteriota; Coriobacteriia; Cc
Bacteria; Bacteroidota; Bacteroidia; Bacteroic
Bacteria: Bacteroidota: Bacteroidia: Bacteroic

Figure 5.13: *Sunburst view.*

5.3 Identify Viral Integration Sites

This tool searches for likely viral/host integration events. The tool works by searching for regions with reads with unaligned ends and/or discordant paired reads, where one read in the pair maps to the host, and the other read maps to a virus.

Notice: this tool can only be used for protocols such as hybrid capture, which specifically enriches for viral genomes while capturing at least some chimeric reads that map to both host and virus genomes.

The approach is the following:

- First, the input reads are mapped simultaneously against the host genome (e.g. human) and a viral database. Internally, the reads are mapped using the 'Find Best References using Read Mapping' tool. Any ambiguous reads are randomly assigned, corresponding to the standard "Non-specific match handling = Map randomly" read mapper option. This produces a host read mapping, and read mappings for all identified viruses. These read mappings are then scanned for potential breakpoints ends, which are the positions showing a pattern of unaligned ends.
- The potential breakpoint ends are filtered based on the following criteria:
  - The number of reads with unaligned ends must be higher than the user-specified criteria
  - The number of reads with unaligned ends must be more than 5% of the maximum for the position with the highest number of unaligned ends for the chromosome/virus.
- For the host, we collect and map all the unaligned ends for a given position against the viral genomes. Then we look at the position where the majority of the reads map on the viral genome, and check if there is a potential breakpoint within 50 bp of that position. Notice: we choose the closest viral breakpoint, and we always choose the read mapping position where the majority of reads map.
- Finally, we look at the broken read pairs on the host genome, where one read was within 500 bp of the host breakpoint (and on the same side as the aligned part of the reads found during the scan for unaligned ends), while the other read in the pair mapped to the virus. If this number of broken reads is larger than a user-specified threshold, the host/virus breakpoint ends are considered a sound match, and we add the host/virus breakpoint to our list of identified breakpoints.

To start the tool, go to:

Toolbox | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Taxonomic Analysis** ( ) | **Identify Viral Integration Sites** ( )

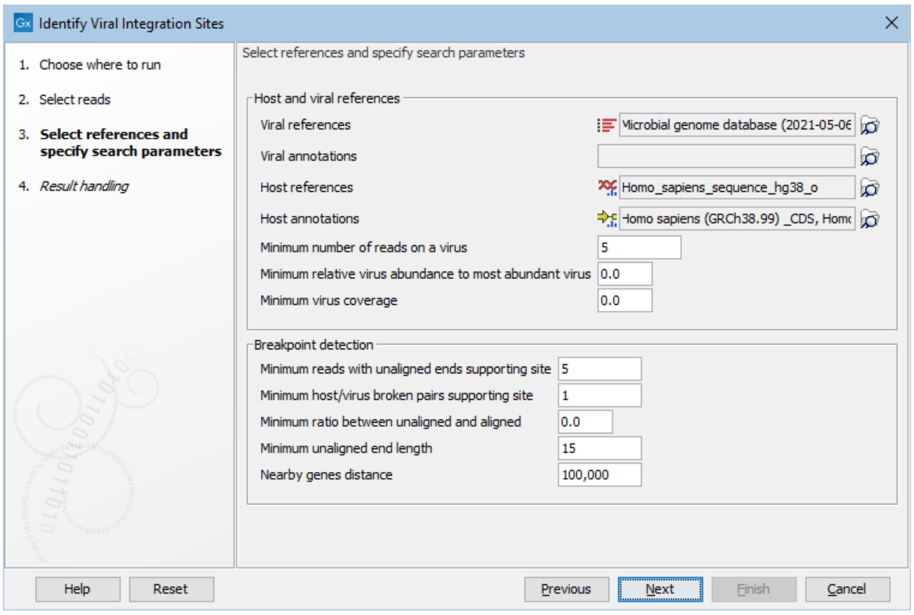
The Identify Viral Integration Sites tool takes one or several single or paired-end read files as input.

After selecting the input reads, it is possible to specify the host and virus references, and adjust the detection parameters, see (figure 5.14).

The following parameters are available:

- **Viral references** The viral sequences. The breakpoints identified from the read mappings against the human reference will be tested against these sequences.
- **Viral annotations** Annotations, such as a Gene or CDS track for the viral sequences. Notice, that these annotations can also be present on the viral sequence input if this is a sequence list. In this case, specifying the annotations here will be used instead of any annotations present on the viral sequence list.
- **Host references** The host sequences.

- **Host annotations** Annotations, such as a Gene or CDS track for the host sequences. Notice, that these annotations can also be present on the host sequence input if this is a sequence list. In that case, specifying the annotations here will overwrite any annotations present on a host sequence list.
- **Minimum number of reads on a virus** At least this many reads must map to a virus before it is included in the analysis.
- **Minimum relative virus abundance to most abundant virus** A reference must have at least this fraction of the reads of the most abundant virus.
- **Minimum virus coverage** The minimum number of nucleotides mapped to the virus reference divided by the reference length before it is included in the analysis.
- **Minimum reads with unaligned ends supporting site** The minimum number of reads required with an unaligned end starting at the same position.
- **Minimum host/virus broken pairs supporting site** The minimum number of paired reads spanning the breakpoint site, where one read maps to the virus, and the other to the host.
- **Minimum ratio between unaligned and aligned** The minimum ratio between reads supporting a breakpoint, and reads with no unaligned ends. This is only checked for the host genome.
- **Minimum unaligned end length** Minimum length of unaligned ends to be considered as supporting a breakpoint.
- **Nearby genes distance** If host genes are located within this distance of an integration event (in basepairs) they are reported in the table view, and in the report.



Select references and specify search parameters

Host and viral references

Viral references	Microbial genome database (2021-05-06
Viral annotations	
Host references	Homo_sapiens_sequence_hg38_o
Host annotations	Homo sapiens (GRCh38.99) _CDS, Homo
Minimum number of reads on a virus	5
Minimum relative virus abundance to most abundant virus	0.0
Minimum virus coverage	0.0

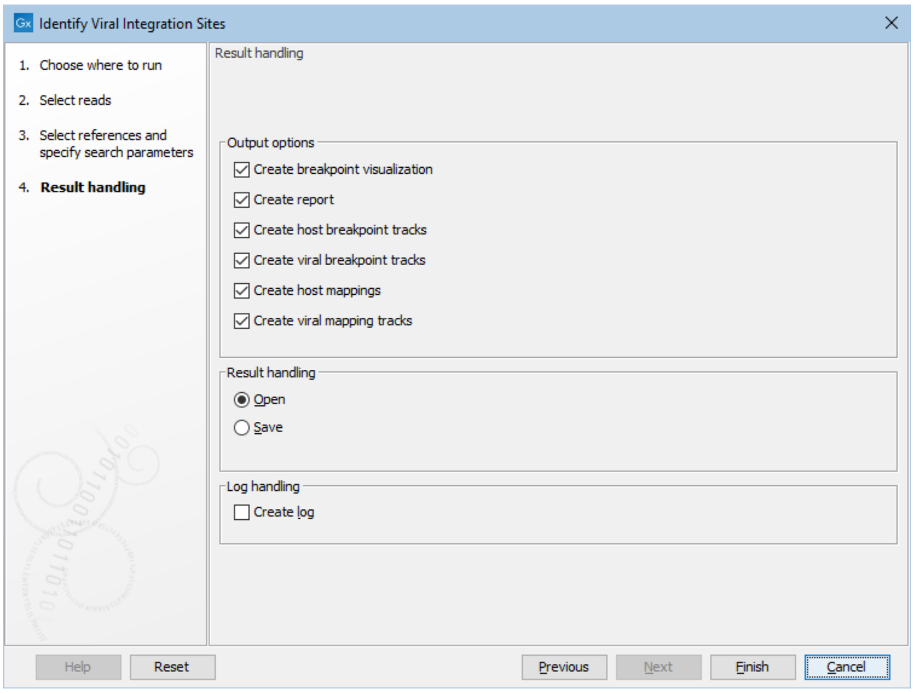
Breakpoint detection

Minimum reads with unaligned ends supporting site	5
Minimum host/virus broken pairs supporting site	1
Minimum ratio between unaligned and aligned	0.0
Minimum unaligned end length	15
Nearby genes distance	100,000

Figure 5.14: *Select references and adjust detection options.*

The final step is to specify the output objects, see (figure 5.15). The following options are available:

- **Create breakpoint visualization** Creates a graphical visualization and a table with breakpoints. This element is explained in more detail in the next section.
- **Create report** Creates a summary report.
- **Create host breakpoint tracks** Creates a feature track with detected breakpoints.
- **Create viral breakpoint tracks** Creates a feature track with detected breakpoints for the identified viruses.
- **Create host mappings** Creates a read mapping for the host references.
- **Create viral mapping tracks** Creates a read mapping for the (detected) viral references.



Result handling

Output options

☑ Create breakpoint visualization

☑ Create report

☑ Create host breakpoint tracks

☑ Create viral breakpoint tracks

☑ Create host mappings

☑ Create viral mapping tracks

Result handling

● Open

○ Save

Log handling

□ Create log

Figure 5.15: *Select output options.*

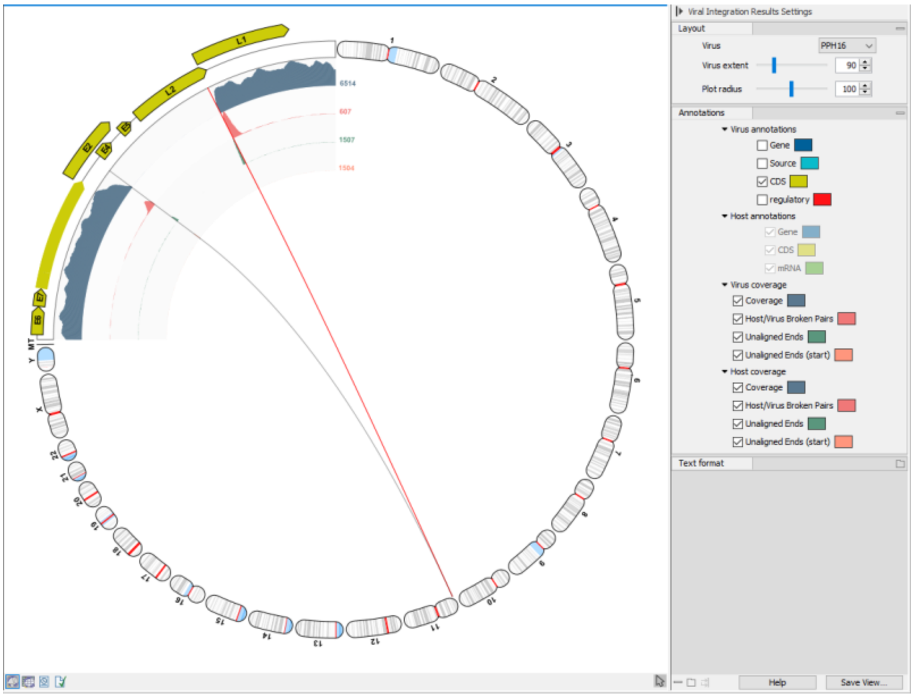
5.3.1 The Viral Integration Viewer

The viral integration viewer presents a graphical view of a virus together with the host genome, shown in a circular plot, see (figure 5.16).

The viral integration viewer is synchronized to the table view: selecting a breakpoint in the table view, will choose the corresponding breakpoint in the graphical view, and vice versa.

The upper left quadrant is a view of the virus. Notice, that if several viruses are detected in a sample, it is possible to choose between them using the **Virus** drop-down in the sidepanel view.

It is possible to zoom in on the host genome, by using the mouse-wheel on a section of the host genome, see (figure 5.17). It is also possible to double-click a breakpoint to zoom in and center on the breakpoint on the host genome.



Viral Integration Results Settings

Layout

Virus: PPH16

Virus extent: 90

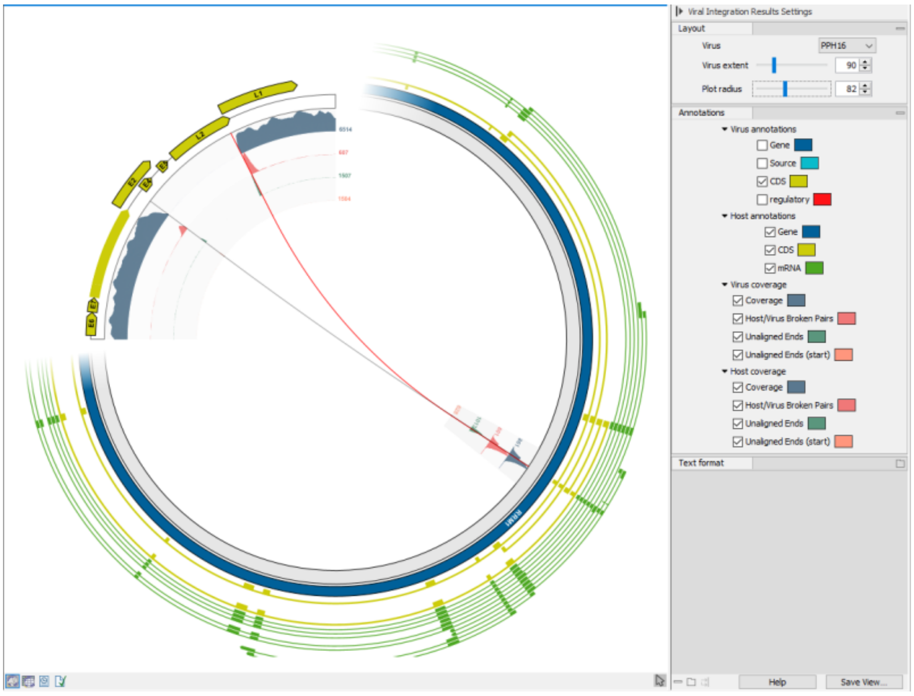
Plot radius: 100

Annotations

- Virus annotations
  - ☐ Gene 🟦
  - ☐ Source 🟦
  - ☑ CDS 🟨
  - ☐ regulatory 🟥
- Host annotations
  - ☑ Gene 🟦
  - ☑ CDS 🟨
  - ☑ mRNA 🟩
- Virus coverage
  - ☑ Coverage 🟦
  - ☑ Host/Virus Broken Pairs 🟥
  - ☑ Unaligned Ends 🟩
  - ☑ Unaligned Ends (start) 🟧
- Host coverage
  - ☑ Coverage 🟦
  - ☑ Host/Virus Broken Pairs 🟥
  - ☑ Unaligned Ends 🟩
  - ☑ Unaligned Ends (start) 🟧

Text format

Figure 5.16: *The viral integration viewer.*



Viral Integration Results Settings

Layout

Virus: PPH16

Virus extent: 90

Plot radius: 82

Annotations

- Virus annotations
  - ☐ Gene 🟦
  - ☐ Source 🟦
  - ☑ CDS 🟨
  - ☐ regulatory 🟥
- Host annotations
  - ☑ Gene 🟦
  - ☑ CDS 🟨
  - ☑ mRNA 🟩
- Virus coverage
  - ☑ Coverage 🟦
  - ☑ Host/Virus Broken Pairs 🟥
  - ☑ Unaligned Ends 🟩
  - ☑ Unaligned Ends (start) 🟧
- Host coverage
  - ☑ Coverage 🟦
  - ☑ Host/Virus Broken Pairs 🟥
  - ☑ Unaligned Ends 🟩
  - ☑ Unaligned Ends (start) 🟧

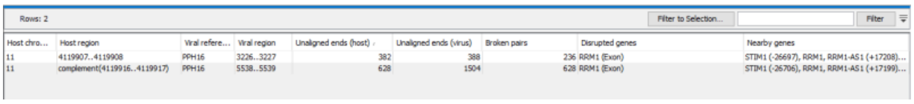
Text format

Figure 5.17: *The viral integration viewer when zoomed in.*

Most elements (coverages, annotations, genome positions, breakpoints) show tooltips with additional information when hovering them with the mouse.

The table view, (figure 5.18), shows summary information for the identified breakpoints, including the host and virus regions, the number of unaligned reads supporting the host and virus end of the breakpoint, and the number of broken host/virus pairs supporting the breakpoint.

The table view also lists any genes that overlap with the breakpoint position. This information is only available if Gene, CDS, or mRNA tracks are provided for the host genome. If CDS and/or mRNA tracks are provided, an additional qualifier ("exon" or "intron") will be added to the output.

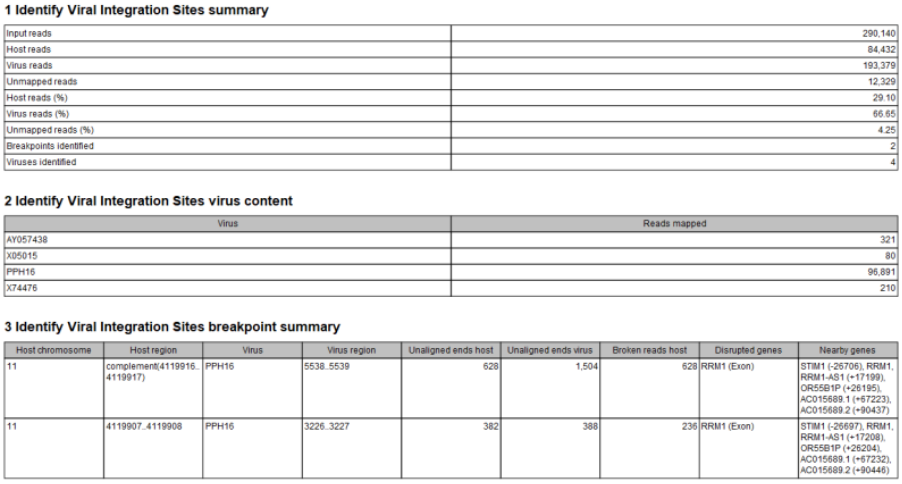


Host chro...	Host region	Viral refere...	Viral region	Unaligned ends (host)	Unaligned ends (virus)	Broken pairs	Disrupted genes	Nearby genes
11	4119907..4119908	PPH16	3226..3227	382	388	236	RRM1 (Exon)	STIM1 (-26697), RRM1, RRM1-AS1 (+17208)...
11	complement(4119916..4119917)	PPH16	5538..5539	628	1504	628	RRM1 (Exon)	STIM1 (-26706), RRM1, RRM1-AS1 (+17199)...

Figure 5.18: *The table view for the detected viral breakpoints.*

5.3.2 The Viral Integration Report

The viral integration report contains an overall summary for the sample. Notice, that reports from multiple samples may be combined using the 'Combine Report' tool.



1 Identify Viral Integration Sites summary

Input reads	290,140
Host reads	84,432
Virus reads	193,379
Unmapped reads	12,329
Host reads (%)	29.10
Virus reads (%)	66.65
Unmapped reads (%)	4.25
Breakpoints identified	2
Viruses identified	4

2 Identify Viral Integration Sites virus content

Virus	Reads mapped
AY057438	321
X05015	80
PPH16	96,891
X74476	210

3 Identify Viral Integration Sites breakpoint summary

Host chromosome	Host region	Virus	Virus region	Unaligned ends host	Unaligned ends virus	Broken reads host	Disrupted genes	Nearby genes
11	complement(4119916..4119917)	PPH16	5538..5539	628	1,504	628	RRM1 (Exon)	STIM1 (-26706), RRM1, RRM1-AS1 (+17199), OR55B1P (+26195), AC015689.1 (+67223), AC015689.2 (+90437)
11	4119907..4119908	PPH16	3226..3227	382	388	236	RRM1 (Exon)	STIM1 (-26697), RRM1, RRM1-AS1 (+17208), OR55B1P (+26204), AC015689.1 (+67232), AC015689.2 (+90446)

Figure 5.19: *The report for the Identify Viral Integration Sites tool.*

Chapter 6

Abundance Analysis

6.1 Merge Abundance Tables

Merge Abundance Tables merges abundance table from different samples. The abundance tables must be of the same type.

For the following types of abundance tables, the tool will merge based on ID or, if no ID is found, on Name:

- OTU tables
- Whole metagenome taxonomic profiling abundance tables
- Functional profile abundance tables
- Resistance abundance tables

Metadata from input tables is transferred to the merged table.

For ASV abundance tables, merging is based on the ASV sequences. To avoid conflicts, taxonomy annotations will be cleared. Use **Assign Taxonomies to Sequences in Abundance Table** to add taxonomies to the merged table, see section 6.2.

To run the Merge Abundance Tables tool:

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Abundance Analysis** () | **Merge Abundance Tables** ()

Select the tables to merge. These must be of the same type (see above).

In the 'Result handling' wizard step, select **Create Report** to generate a report with summary information on input and output abundance tables.

6.2 Assign Taxonomies to Sequences in Abundance Table

The Assign Taxonomies to Sequences in Abundance Table tool lets you add taxonomies to abundance table features that have sequences associated. This is useful for annotating amplicon sequence variant (ASV) tables, and OTU tables with de novo OTUs where sequences are not annotated by the initial analysis tools.

The tool requires a reference index and works by mapping each sequence from the abundance table to this reference index. The underlying analysis is the same as for **Taxonomic Profiling**, see section 5.2.

Creating the required reference index

You create a reference index using **Create Taxonomic Profiling Index**, see section 15.4. As input, you will need a reference database, i.e. a sequence list containing reference sequences with taxonomy annotations.

Reference database can be obtained using one of the Download Database tools. The choice of reference database depends on your data.

For amplicon data, consider the reference databases available with **Download Amplicon-Based Reference Database**, see section 14.1.

For whole genome data, you may use the databases from **Download Curated Microbial Reference Database**, see section 15.1. Alternatively, create your own reference database with **Download Custom Microbial Reference Database**, see section 15.2.

Running the tool

To run the Assign Taxonomies to Sequences in Abundance Table tool:

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Abundance Analysis** () | **Assign Taxonomies to Sequences in Abundance Table** ()

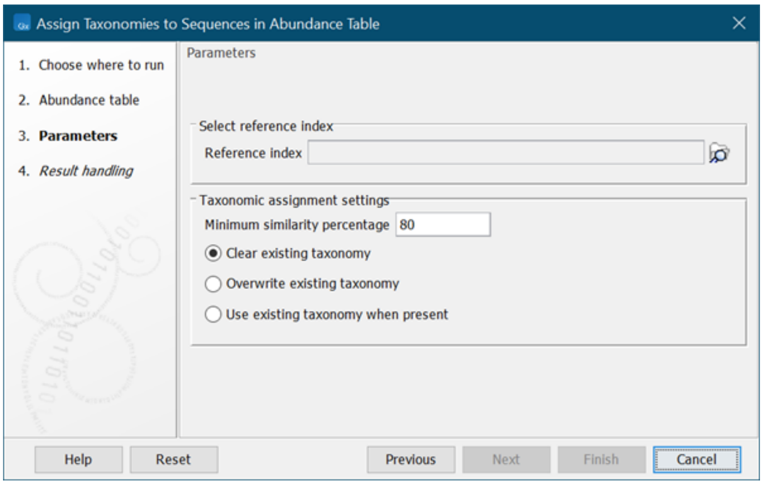
Select the abundance table with sequences to be annotated.

Select the reference index to map the abundance table sequences to (figure 6.1).

Choose settings for taxonomic assignment:

- **Minimum similarity percentage** Sequences in the abundance table must be at least this similar to sequence in the reference index to be matched and get a new taxonomy assigned.
- **Clear existing taxonomy** All existing abundance table taxonomy annotations are removed. Only abundance table sequences with a reference index match will get a taxonomy assignment.
- **Overwrite existing taxonomy** Abundance table sequences with a reference index match will get a new taxonomy assigned. Sequences with no match will retain the existing taxonomy annotation.
- **Use existing taxonomy when present** Only abundance table sequences that do not already have a taxonomic annotation will get a new taxonomy assigned.

Select **Create Report** to generate a report with summary information on taxonomic assignment.



Assign Taxonomies to Sequences in Abundance Table

1. Choose where to run
2. Abundance table
3. **Parameters**
4. *Result handling*

Parameters

Select reference index

Reference index

Taxonomic assignment settings

Minimum similarity percentage 80

● Clear existing taxonomy

○ Overwrite existing taxonomy

○ Use existing taxonomy when present

Figure 6.1: *Select reference index and set parameters for taxonomic assignment.*

The output

- **Abundance table with taxonomy annotations** The new taxonomy assignments are listed in the **Taxonomy** column.
- **Assign taxonomies report** The report contains the following sections:
  - **Summary** Information on the sequences in the abundance table.
  - **Reference index summary** Information on the reference index.
  - **Taxonomy assignment**
    - **Sequences with reference index match**: Sequences that met the Minimum similarity treshold.
      - **Taxonomy assigned (was blank)**: Taxonomy was blank, new taxonomy has been assigned.
      - **Taxonomy updated**: The existing taxonomy has been replaced.
      - **Existing taxonomy retained**: The existing taxonomy is retained. This can happen when the taxonomy of the matched reference index sequence is identical to the existing taxonomy, or when *Taxonomy assignment* was set to *Use existing taxonomy when present*.
    - **Sequences with no reference index match (insufficient similarity)**: Sequences that did not meet the Minimum similarity threshold.
      - **Existing taxonomy retained**: Sequences for which an existing taxonomy remains.
      - **No taxonomy**: Sequences with no taxonomy.

6.3 Alpha Diversity

Alpha diversity is the diversity within a particular area or ecosystem, usually expressed by the number of species (i.e., species richness) in that ecosystem. Alpha diversity estimates are calculated from a series of rarefaction analyses and hence dependent on sampling depth.

The Alpha Diversity tool takes abundance tables as input. Abundance tables can be generated in the workbench by various tools, for example: OTU clustering, Build Functional Profile and Taxonomic Profiling. With the first two tools, the abundance tables generated are count-based, and Alpha diversity measures calculated from such tables give an absolute number of species. However, when using an abundance table generated by e.g. the Taxonomic Profiling tool, Alpha diversity results will not give an absolute number of species, but rather estimates that are useful for comparative studies, i.e., to assess the depth of sequencing, or to compare different communities.

To run the tool, go to:

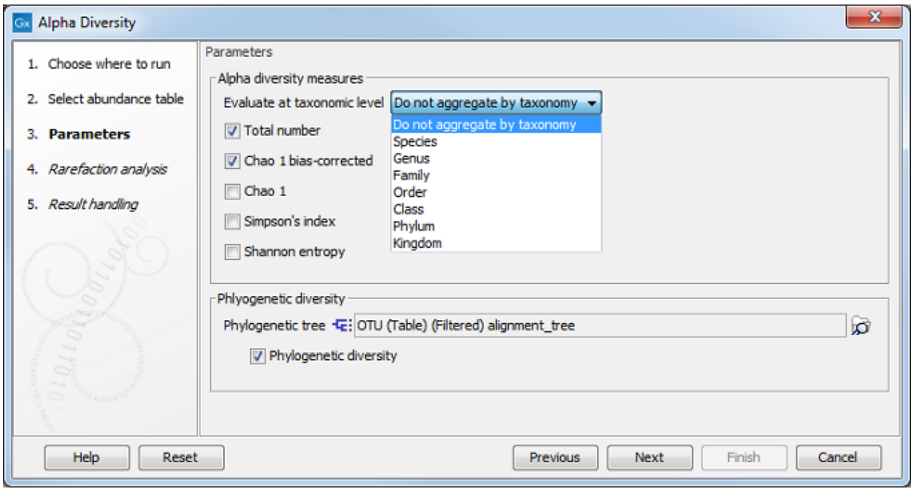
> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **Alpha Diversity** ( )

Choose an abundance table to use as input.

The next wizard window offers you to set up different analysis parameters (figure 6.2).

For example, you can calculate metrics at a specific taxonomic level: the tool will then aggregate the features by taxonomy (so that OTUs from the same phylum will be grouped together) before computing the metric. The default value is to not aggregate by taxonomy. You then select which diversity measures to calculate (see section 6.3.1).

If you are working with OTU abundance tables, you can also specify an appropriate phylogenetic tree for computing phylogenetic diversity. In that case, you must have aligned the OTUs and constructed a phylogeny before running the Alpha Diversity tool. Note that the "Evaluate at taxonomic level" option described above does not apply to the "Phylogenetic Diversity" metric, since that metric is not using taxonomic information, but is making use of a phylogenetic tree based on OTU sequences.



Parameters

Alpha diversity measures

Evaluate at taxonomic level: Do not aggregate by taxonomy

☑ Total number

☑ Chao 1 bias-corrected

□ Chao 1

□ Simpson's index

□ Shannon entropy

Phlyogenetic diversity

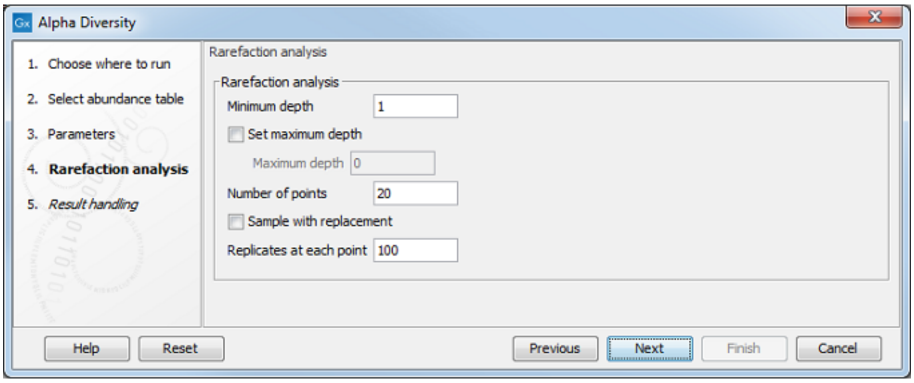
Phylogenetic tree: OTU (Table) (Filtered) alignment_tree

☑ Phylogenetic diversity

Figure 6.2: *Set up parameters for the Alpha Diversity tool.*

In the following dialog (figure 6.3), set up the rarefaction analysis parameters.

The rarefaction analyses are done differently depending on the type of abundance table used as input. For abundance tables where abundances are counts, such as OTU and functional abundance tables, rarefaction is calculated by sub-sampling the abundances in the different samples to the same depths. For taxonomic profiling abundance tables, where abundances



Rarefaction analysis

Rarefaction analysis

Minimum depth 1

☐ Set maximum depth

Maximum depth 0

Number of points 20

☐ Sample with replacement

Replicates at each point 100

Figure 6.3: *Set up parameters for the Rarefaction analysis.*

are coverage estimates, sub-sampling is not possible. Instead, diversity is estimated using a probabilistic model corresponding to our qualification criteria (see section 5.2).

The rarefaction analysis parameters will define the granularity of the alpha diversity curve.

- **Minimum depth to sample** is set to 1 by default.
- **Maximum depth to sample** If this option is not checked, the maximum depth is set it to the total number of reads (in the case of one sample) or the total number of reads of the sample with most reads.
- **Numbers of points** Number of different depths to be sampled. For example, if you choose to sample 5 depths between 1000 and 5000, the algorithm will sub-sample each sample at 1000, 2000, 3000, 4000, and 5000 reads.
- **Sample with replacement** Whether the sampling should be performed with or without replacement.
- **Replicates at each depth** (for counts-based abundance tables only). How many times the algorithm sub-samples the data at each depth.

The tool will generate a graph for each selected Alpha diversity measure (figure 6.4). Using the Lines and dots editor on the right hand side panel, it is possible to color samples according to groups defined by associated metadata. Note that you can filter metadata by typing the appropriate text in the field above each list of metadata elements. This is an easy way to change the visualization of a group of data at once.

Note that the option "Show derived legend info" is enabled by default (figure 6.5). According to this setting, the legend(s) for which metadata categories happen to be "shared" for all items in the legend will display the dependencies between the different categories. In this example, the "Location" category determines Dot Type, and the "Antibiotic" category determines Line Color. For this particular data set, all samples with a specific location have the same antibiotic resistance. The "Show derived legend info" option enables the legends to show such implicit dependencies in the data. If such a visualization is not wished for, the option can be disabled, and the legend will show only the metadata category values that were explicitly selected in the right hand side panel.

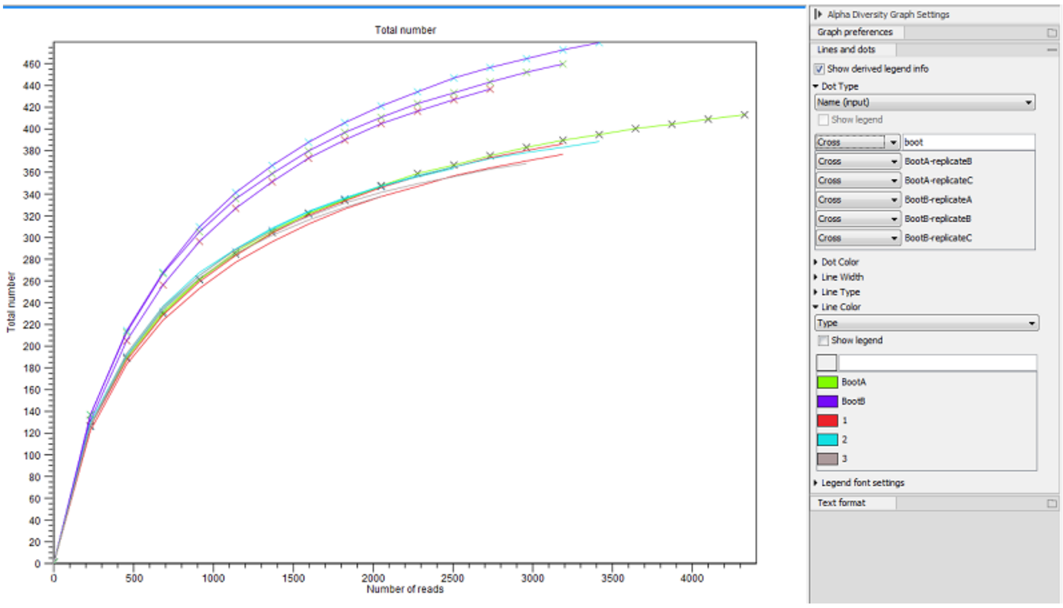


Figure 6.4: *An example of Alpha Diversity graph based on phylogenetic diversity.*

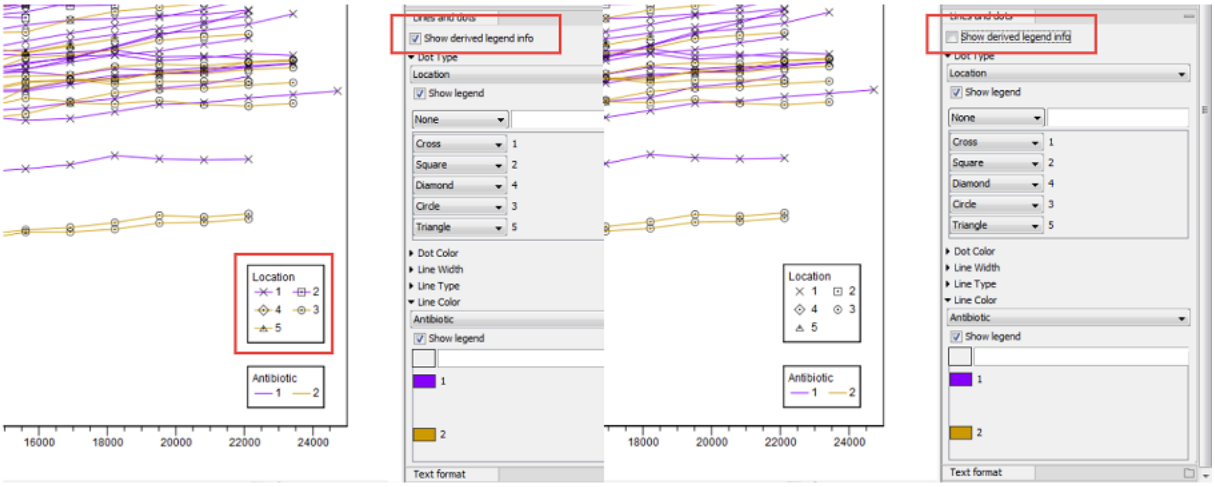
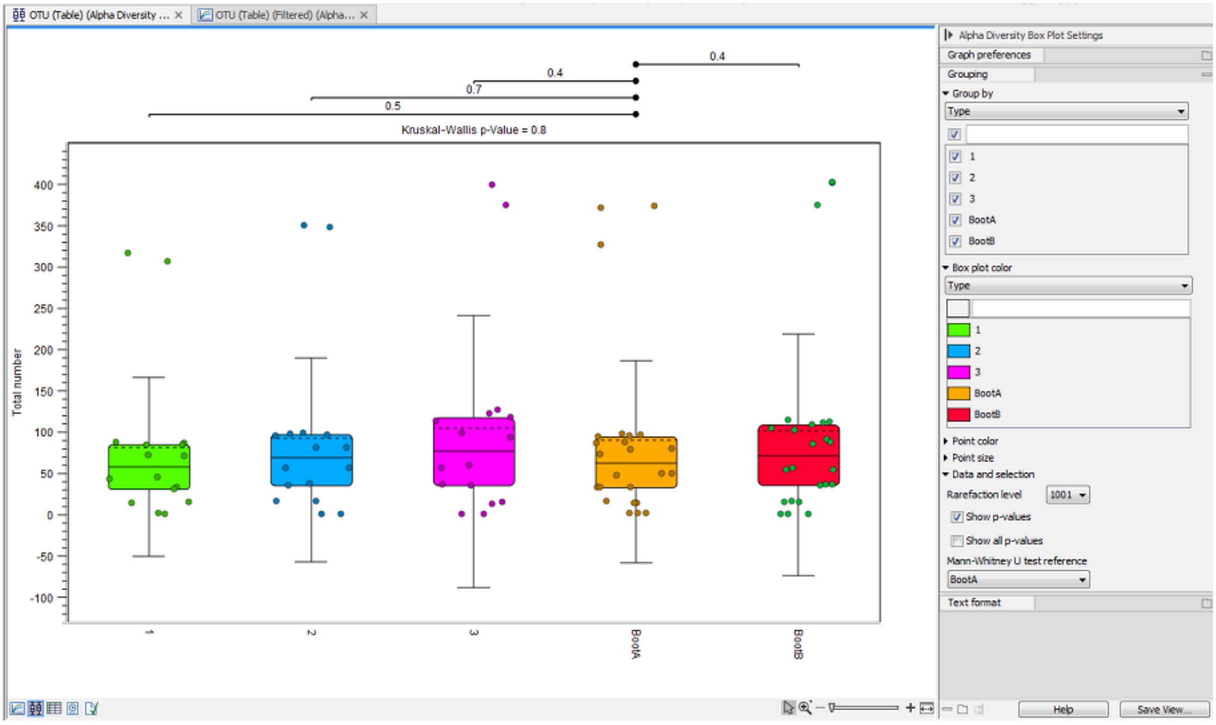


Figure 6.5: *Example of the difference between having the "Show derived legend info" enabled or disabled. When enabled, the legend helps visualize that "location" and "antibiotic" are dependent for this particular data set.*

It is also possible to view the Alpha diversity measures as Box plot to see if samples of a certain group are significantly different than those of other groups (figure 6.6). For example, one can check if soils of a certain type contain more bacterial species than other samples.

The box plot view can also display the following statistics:

- **Rarefaction level** This drop down menu allows to choose which value of the rarefaction curve should be used. The values of "Rarefaction level" are the same as the horizontal axis in the Alpha Diversity Graph view and correspond to the depths at which the input data is sub-sampled before calculating the diversity metric in the Alpha Diversity tool.
- **Kruskal-Wallis H test** This test is used to assess whether the values originate from the


OTU (Table) (Alpha Diversity ...
OTU (Table) (Filtered) (Alpha...
0.4
0.4
0.7
0.5
Kruskal-Wallis p-Value = 0.8
Total number
400
350
300
250
200
150
100
50
0
-50
-100
1
2
3
BootA
BootB
Help
Save View...

Alpha Diversity Box Plot Settings

Graph preferences

Grouping

Group by: Type

☑ 1
☑ 2
☑ 3
☑ BootA
☑ BootB

Box plot color: Type

🟩 1
🟦 2
🟪 3
🟧 BootA
🟥 BootB

Point color

Point size

Data and selection

Rarefaction level: 1001

☑ Show p-values

☐ Show all p-values

Mann-Whitney U test reference: BootA

Text format

Figure 6.6: *Alpha diversities shown in a box plot.*

same distribution or whether their distribution is different depending on the group they belong to. This test is a nonparametric alternative to ANOVA (i.e., it does not depend on the data following a given distribution - the normal distribution in case of ANOVA). A significant p-value for the Kruskal-Wallis test means that at least one group follows a different distribution, but does not specify which pairs have different distributions.

- **Mann-Whitney U test** The tool therefore performs a pair-wise **Mann-Whitney U test** to specifically determine which pairs of groups follow different distributions. These statistical tests are performed when the "Show p-values" option is checked. If the "Show all p-values" is checked, all pairwise Mann-Whitney U tests are performed, while when it is not checked, only the pairs that contain the reference group specified in the "Mann-Whitney U test reference" option are considered.

6.3.1 Alpha diversity measures

The available diversity measures are:

- Total number: The number of features (e.g. OTUs when doing OTU clustering, GO terms when building functional profiles or organisms when performing taxonomic profiling) observed in the sample.
- Chao 1 bias-corrected: Chao1-bc $= D + \frac{f_1(f_1-1)}{2(f_2+1)}$.
- Chao 1: Chao1 $= D + \frac{f_1^2}{2f_2}$.
- Simpson's index: SI $= 1 - \sum_{i=1}^{n} p_i^2$.

- Shannon entropy: H $= \sum_{i=1}^{n} p_i \log_2 p_i$.

where n is the number of features; D is the number of distinct features observed in the sample; f_1 is the number of features for which only one read has been found in the sample; f_2 is the number of features for which two reads have been found in the sample; and p_i is the fraction of reads that belong to feature i.

Note that Chao-based methods deal with singletons and doubletons, i.e., rows with exactly one or two reads (counts) associated. These measures are thus not available for whole metagenome taxonomic profiles that are characterized by coverage estimate.

The following distances are also available:

- Phylogenetic diversity: $PD = \sum_{i=1}^{n} b_i I(p_i > 0)$

where n is the number of branches in the phylogenetic tree, b_i is the length of branch i; p_i is the proportion of taxa descending from branch i; and the indicator function $I(p_i > 0)$ and $I(p_i^B > 0)$ assumes the value of 1 if any taxa descending from branch i is present in the sample or 0 otherwise.

6.4 Beta Diversity

Beta diversity examines the change in species diversity between ecosystems. The analysis is done in two steps. First, the tool estimates a distance between each pair of samples (see section 6.4.1). Once the distance matrix is calculated, the beta diversity analysis tool performs Principal Coordinate Analysis (PCoA) on the distance matrices. These can be visualized by selecting the PCoA icon ( ) in the bottom of the Beta Diversity results ( ).

The Beta Diversity tool takes abundance tables as input. Abundance tables can be generated in the workbench by various tools, for example: OTU clustering, Build Functional Profile and Taxonomic Profiling.

If you are working with an OTU table, you can specify an appropriate phylogenetic tree for computing phylogenetic diversity. In that case, you must have aligned the OTUs and constructed a phylogeny before running the Beta Diversity tool.

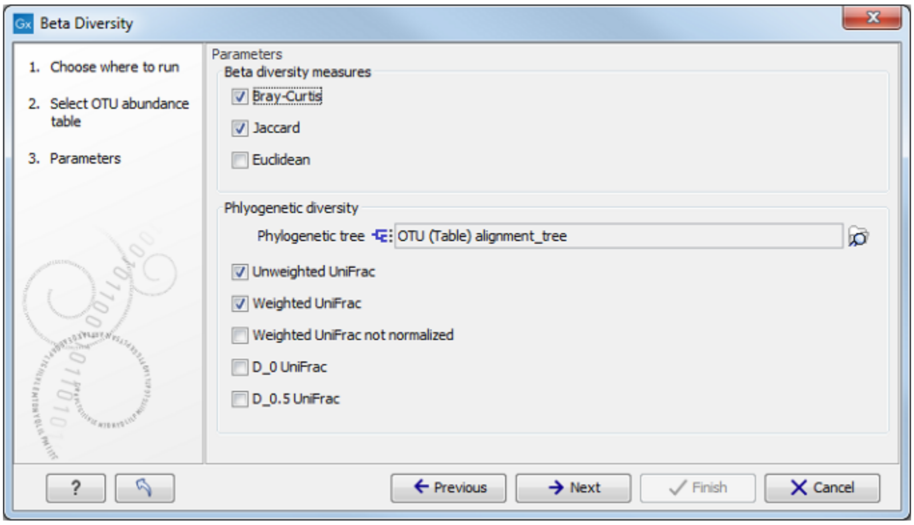
To run the tool, open

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **Beta Diversity** ( )

Select an abundance table with more than one sample as input (e.g., a multi-sample OTU or merged abundance table) and set the parameters for the beta diversity analysis as shown in figure 6.7.

The output of the tool is a 3D PCoA plot (figure 6.8) that can also be seen as a table or a 2D Principal Coordinate Plot.

> If you have problems viewing the 3D plot, please check your system matches the requirements for 3D viewers. See https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=System_requirements.html.



Beta Diversity

1. Choose where to run
2. Select OTU abundance table
3. Parameters

Parameters

Beta diversity measures

☑ Bray-Curtis

☑ Jaccard

□ Euclidean

Phlyogenetic diversity

Phylogenetic tree: OTU (Table) alignment_tree

☑ Unweighted UniFrac

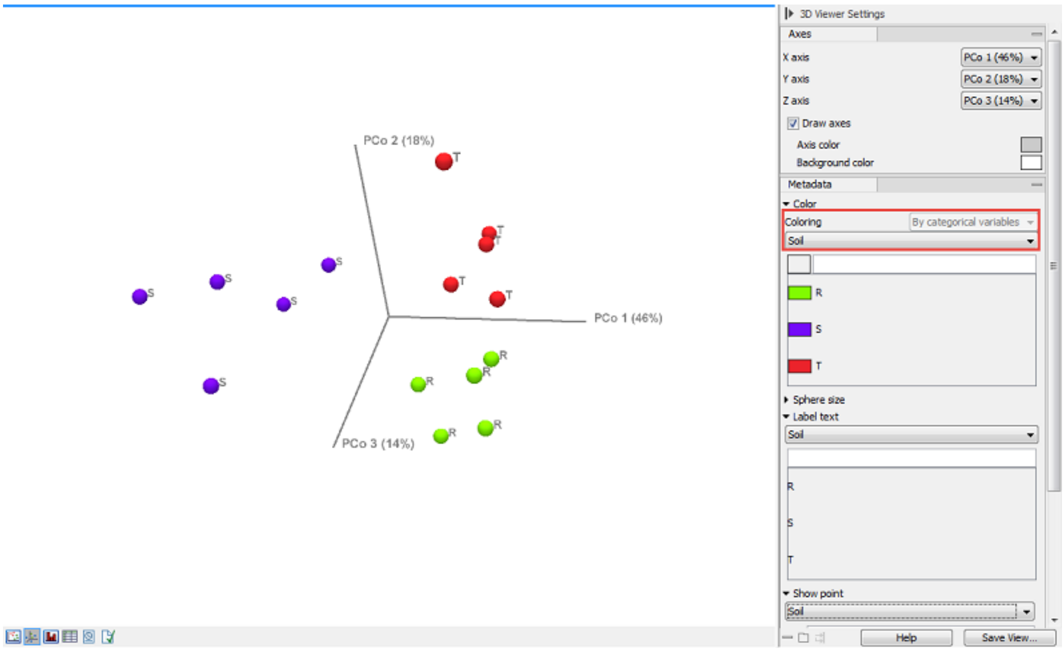
☑ Weighted UniFrac

□ Weighted UniFrac not normalized

□ D_0 UniFrac

□ D_0.5 UniFrac

Figure 6.7: *Set up parameters for the Beta diversity tool.*



3D Viewer Settings

Axes

X axis: PCo 1 (46%)

Y axis: PCo 2 (18%)

Z axis: PCo 3 (14%)

☑ Draw axes

Axis color

Background color

Metadata

Color

Coloring: By categorical variables

Soil

🟩 R

🟪 S

🟥 T

Sphere size

Label text

Soil

R

S

T

Show point

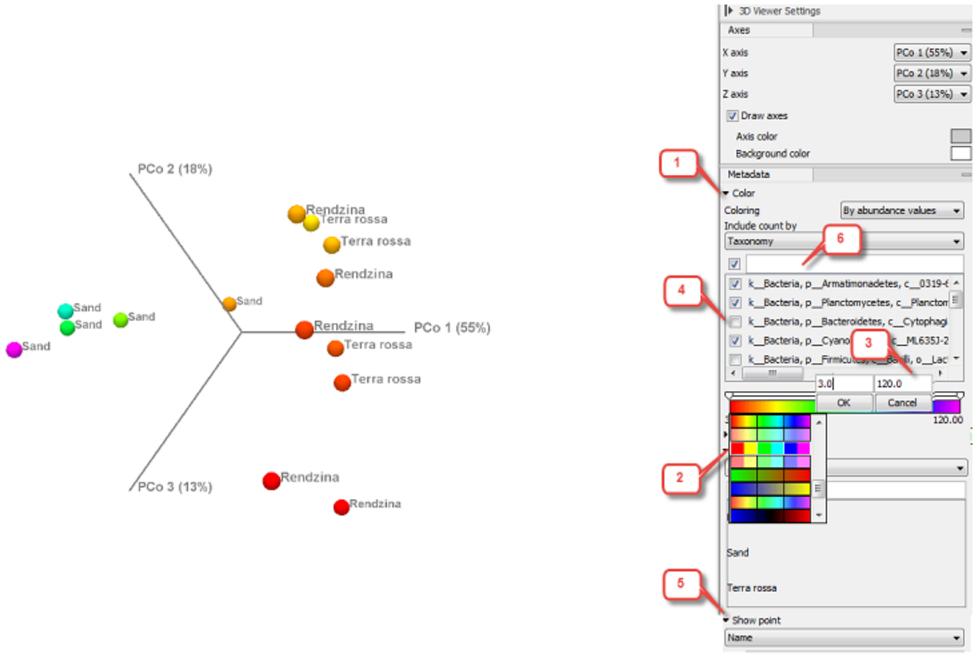
Soil

Figure 6.8: *Beta diversity results seen as a 3D PCoA, with coloring done according to the categories dedined by the metadata.*

Use the settings in the right hand Side Panel of the PCoA (2D or 3D) to modify the plot visualization.

In the section Metadata, the **Color** menu (1) allows you to choose whether you want your data to be colored according to categorical variables (the ones defined by the metadata, as seen in figure 6.8) or by abundance values (figure 6.9).

Coloring per abundance values is done with a gradient scheme. Click on the gradient bar to choose from several color schemes (2). Double-click on the slider to set specific values for the gradient (3).



3D Viewer Settings

Axes

X axis: PCo 1 (55%)

Y axis: PCo 2 (18%)

Z axis: PCo 3 (13%)

☑ Draw axes

Axis color

Background color

Metadata

Color

Coloring: By abundance values

Include count by: Taxonomy

☑ k__Bacteria, p__Armatimonadetes, c__0319-6

☑ k__Bacteria, p__Planctomycetes, c__Planctom

□ k__Bacteria, p__Bacteroidetes, c__Cytophagi

☑ k__Bacteria, p__Cyano... c__ML635J-2

□ k__Bacteria, p__Firmicu... Bacilli, o__Lac

3.0 120.0

Sand

Terra rossa

Show point

Name

Figure 6.9: *Beta diversity results seen as a 3D PCoA, with coloring done according to taxonomic abundance values.*

When coloring "By abundance values", it is possible to control the abundance value calculation through existing metadata fields such as Name, EC numbers, or Taxonomy (depending on the type of abundance table input). The drop down menu will then display the items that can be deselected (4) if you want to remove them from the abundance value calculation: this will not remove any of the data point from the PCoA view, but just change the abundance values of the affected point and therefore its coloring. To remove a data point from the plot, use the **Show point** menu (5) below in the Side Panel.

As in other metadata Side Panel features, use the field above each menu (6) to filter that menu. This is particularly helpful when menus are very long, as is the case with taxonomies for example.

6.4.1 Beta diversity measures

The following beta diversity measures are available:

- Bray-Curtis: $B = \frac{\sum_{i=1}^{n} |x_i^A - x_i^B|}{\sum_{i=1}^{n} (x_i^A + x_i^B)}$
- Jaccard: $J = 1 - \frac{\sum_{i=1}^{n} \min(x_i^A, x_i^B)}{\sum_{i=1}^{n} \max(x_i^A, x_i^B)}$
- Euclidean: $E = \sum_{i=1}^{n} \sqrt{(x_i^A - x_i^B)^2}$

where n is the number of OTUs and x_i^A and x_i^B are the abundances of OTU i in samples A and B, respectively.

The following distances are also available:

- Unweighted UniFrac: $d^{(U)} = \frac{\sum\limits_{i=1}^{n} b_i \left| I(p_i^A>0)-I(p_i^B>0) \right|}{\sum\limits_{i=1}^{n} b_i}$
- Weighted UniFrac: $d^{(W)} = \frac{\sum\limits_{i=1}^{n} b_i \left| p_i^A - p_i^B \right|}{\sum\limits_{i=1}^{n} b_i (p_i^A + p_i^B)}$
- Weighted UniFrac not normalized: $d^{(w)} = \sum\limits_{i=1}^{n} b_i \left| p_i^A - p_i^B \right|$
- D_0 UniFrac: The generalized UniFrac distance $d^{(0)} = \frac{\sum\limits_{i=1}^{n} b_i \left| \frac{p_i^A - p_i^B}{p_i^A + p_i^B} \right|}{\sum\limits_{i=1}^{n} b_i}$
- D_0.5 UniFrac: The generalized UniFrac distance $d^{(0.5)} = \frac{\sum\limits_{i=1}^{n} b_i \sqrt{p_i^A + p_i^B} \left| \frac{p_i^A - p_i^B}{p_i^A + p_i^B} \right|}{\sum\limits_{i=1}^{n} b_i \sqrt{p_i^A + p_i^B}}$

where n is the number of branches in the phylogenetic tree, b_i is the length of branch i; p_i^A and p_i^B are the proportion of taxa descending from branch i for samples A and B, respectively; and the indicator functions $I(p_i^A > 0)$ and $I(p_i^B > 0)$ assume the value of 1 if any taxa descending from branch i is present in samples A and B, respectively, or 0 otherwise.

The unweighted UniFrac distance gives comparatively more importance to rare lineages, while the weighted UniFrac distance gives more important to abundant lineages. The generalized UniFrac distance $d^{(0.5)}$ offers a robust tradeoff [Chen et al., 2012].

6.5 PERMANOVA Analysis

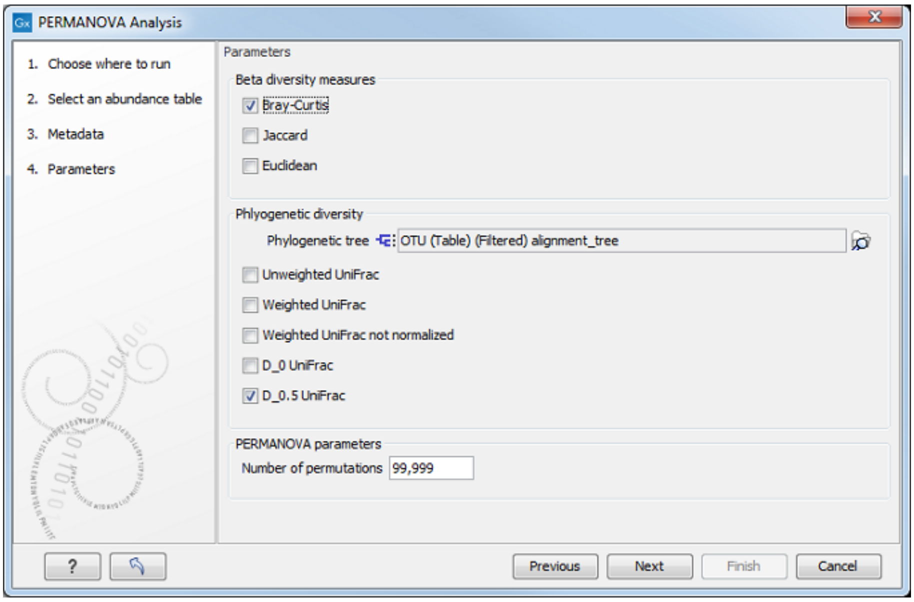
PERMANOVA Analysis (PERmutational Multivariate ANalysis Of VAriance, also known as non-parameteric MANOVA [Anderson, 2001]), can be used to measure effect size and significance on beta diversity for a grouping variable. For example, it can be used to show whether OTU abundance profiles of replicate samples taken from different locations vary significantly according to the location or not. The significance is obtained by a permutation test.

To perform a PERMANOVA analysis, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **PERMANOVA Analysis** ( )

Choose an abundance table with more than one sample as input (e.g., a multi-sample OTU or merged abundance table) and specify the metadata group you would like to test. You will need more than one replicate in the metadata group you select.

In the "Parameters" dialog (figure 6.10), you can choose which Beta diversity measure to use (see section 6.4.1). If you are working with OTU abundance tables, you can also specify in the next dialog the phylogenetic tree reconstructed from the alignment of the most abundant OTUs and the phylogenetic diversity measures you wish to use for this analysis. Finally, choose how many permutations should be performed (the default is set to 99,999).



Parameters

Beta diversity measures

☑ Bray-Curtis

□ Jaccard

□ Euclidean

Phlyogenetic diversity

Phylogenetic tree OTU (Table) (Filtered) alignment_tree

□ Unweighted UniFrac

□ Weighted UniFrac

□ Weighted UniFrac not normalized

□ D_0 UniFrac

☑ D_0.5 UniFrac

PERMANOVA parameters

Number of permutations 99,999

Figure 6.10: *Beta diversity and Phylogenetic diversity measures are included in the PERMANOVA analysis.*

The output of the analysis is a report which contains two tables for each beta diversity measure used:

- A table showing the metadata variable used, its groups and the results of the test (pseudo-f-statistic and p-value)
- A PERMANOVA analysis for each pair of groups and the results of the test (pseudo-f-statistic and p-value). Bonferroni-corrected p-values (which correct for multiple testing) are also shown.

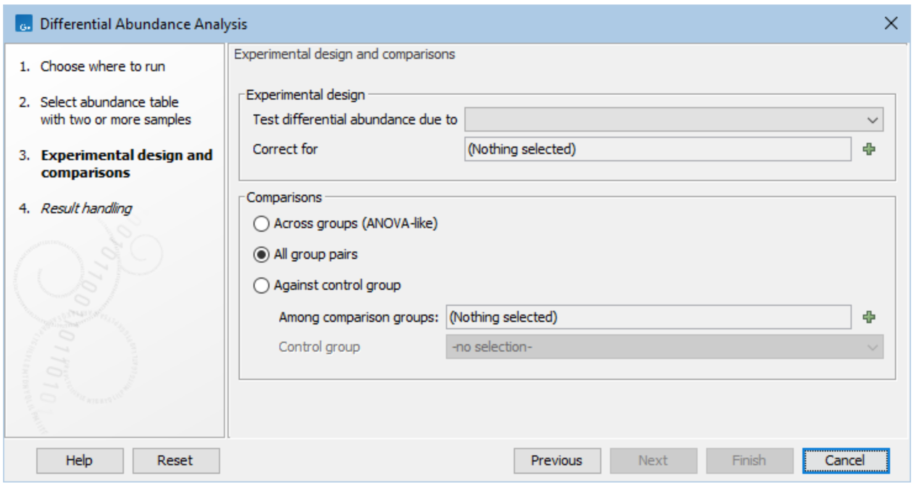
6.6 Differential Abundance Analysis

This tool performs a generalized linear model differential abundance test on samples, or groups of samples defined by metadata. The tool models each feature (e.g., an OTU, an organism or species name or a GO term) as a separate Generalized Linear Model (GLM), where, after performing TMM normalization, it is assumed that abundances follow a Negative Binomial distribution. The Wald test is used to determine significance between group pairs, whereas a Likelihood Ratio test is used in the Across groups (ANOVA-like) comparison. The underlying statistical model is the same as the one used by the Differential Expression for RNA-Seq tool described in details here: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Differential_Expression.html.

To run the tool:

Toolbox | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **Differential Abundance Analysis** ( )

Select an abundance table with more than one sample as input (e.g., a multi-sample OTU or merged abundance table), and specify if you want to test differential abundance based on metadata defined groups of samples (figure 6.11). It is also possible to correct the results based on a metadata defined group of samples. Finally you can choose whether you want the comparison to be done across groups, between all group pairs or against a control group.


Differential Abundance Analysis
1. Choose where to run
2. Select abundance table with two or more samples
3. **Experimental design and comparisons**
4. *Result handling*
Help
Reset
Previous
Next
Finish
Cancel

Experimental design and comparisons

Experimental design

Test differential abundance due to

Correct for (Nothing selected)

Comparisons

○ Across groups (ANOVA-like)

● All group pairs

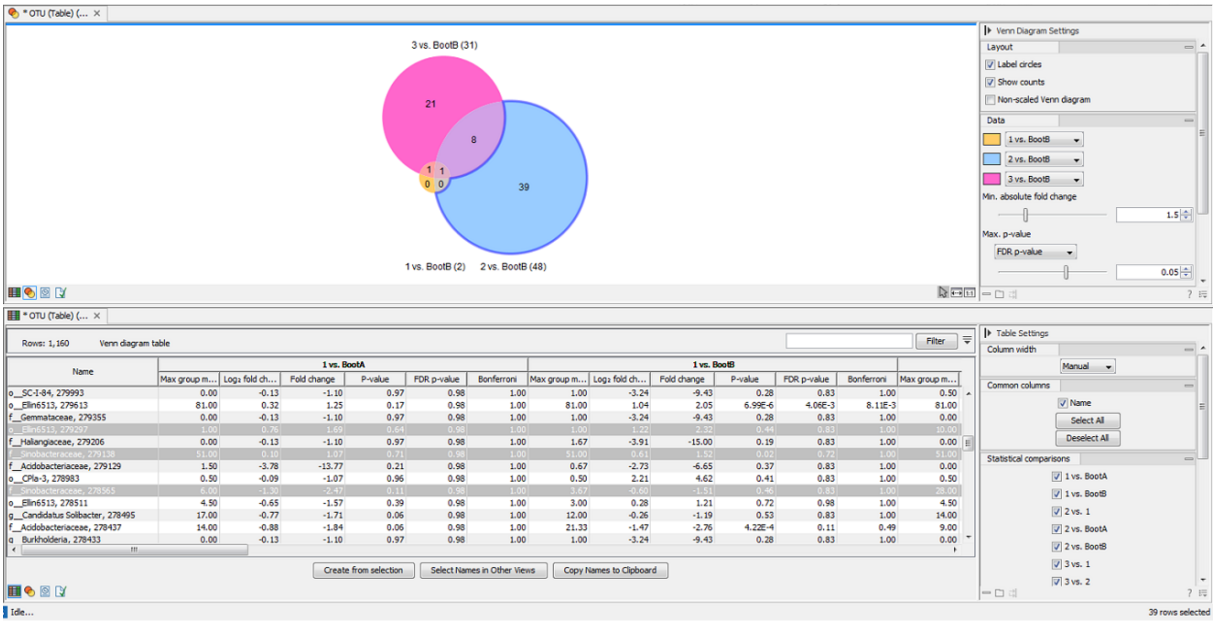
○ Against control group

Among comparison groups: (Nothing selected)

Control group -no selection-

Figure 6.11: *Specify an abundance table and all other parameters.*

The tool generates a Venn diagram for three pairwise comparisons at a time (figure 6.12). You can select which comparisons should be shown using the drop down menus in the side panel. Clicking a circle segment in the Venn diagram will select the samples of this segment in the differential abundance analysis table view. The table summarizes abundances, fold changes, differential abundance p-values, multi-sample corrected p-values, etc.



Venn Diagram Settings

Layout

☑ Label circles

☑ Show counts

□ Non-scaled Venn diagram

Data

🟨 1 vs. BootB

🟦 2 vs. BootB

🟪 3 vs. BootB

Min. absolute fold change 1.5

Max. p-value FDR p-value 0.05

Table Settings

Column width Manual

Common columns

☑ Name

Select All

Deselect All

Statistical comparisons

☑ 1 vs. BootA

☑ 1 vs. BootB

☑ 2 vs. 1

☑ 2 vs. BootA

☑ 2 vs. BootB

☑ 3 vs. 1

☑ 3 vs. 2

Rows: 1,160 Venn diagram table

Name	1 vs. BootA						1 vs. BootB						
	Max group m...	Log₂ fold ch...	Fold change	P-value	FDR p-value	Bonferroni	Max group m...	Log₂ fold ch...	Fold change	P-value	FDR p-value	Bonferroni	Max group m...
o__SC-I-84, 279993	0.00	-0.13	-1.10	0.97	0.98	1.00	1.00	-3.24	-9.43	0.28	0.83	1.00	0.50
o__Ellin6513, 279613	81.00	0.32	1.25	0.17	0.98	1.00	81.00	1.04	2.05	6.99E-6	4.06E-3	8.11E-3	81.00
f__Gemmataceae, 279355	0.00	-0.13	-1.10	0.97	0.98	1.00	1.00	-3.24	-9.43	0.28	0.83	1.00	0.00
o__Ellin6513, 279297	1.00	0.76	1.69	0.64	0.98	1.00	1.00	1.22	2.32	0.44	0.83	1.00	10.00
f__Haliangiaceae, 279206	0.00	-0.13	-1.10	0.97	0.98	1.00	1.67	-3.91	-15.00	0.19	0.83	1.00	0.00
f__Sinobacteraceae, 279138	51.00	0.10	1.07	0.71	0.98	1.00	51.00	0.61	1.52	0.02	0.72	1.00	51.00
f__Acidobacteriaceae, 279129	1.50	-3.78	-13.77	0.21	0.98	1.00	0.67	-2.73	-6.65	0.37	0.83	1.00	0.00
o__CPla-3, 278983	0.50	-0.09	-1.07	0.96	0.98	1.00	0.50	2.21	4.62	0.41	0.83	1.00	0.50
f__Sinobacteraceae, 278565	6.00	-1.30	-2.47	0.11	0.98	1.00	3.67	-0.60	-1.51	0.46	0.83	1.00	28.00
o__Ellin6513, 278511	4.50	-0.65	-1.57	0.39	0.98	1.00	3.00	0.28	1.21	0.72	0.98	1.00	4.50
g__Candidatus Solibacter, 278495	17.00	-0.77	-1.71	0.06	0.98	1.00	12.00	-0.26	-1.19	0.53	0.83	1.00	14.00
f__Acidobacteriaceae, 278437	14.00	-0.88	-1.84	0.06	0.98	1.00	21.33	-1.47	-2.76	4.22E-4	0.11	0.49	9.00
g Burkholderia, 278433	0.00	-0.13	-1.10	0.97	0.98	1.00	1.00	-3.24	-9.43	0.28	0.83	1.00	0.00

39 rows selected

Figure 6.12: *Venn diagram for three comparisons at a time. Selecting a segment will highlight the samples in the differential abundance table opened below in split view.*

The values included in the table for each pairwise comparison are:

- **Max group means** For each group in the statistical comparison, the average measured abundance or expression value is calculated. The Max Groups Means is the maximum of the average values.

- **log2 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). 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.
- **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.

It is possible to create a subset list of samples using the Create from selection button. As usual, the table can be adjusted with the right hand side panel options: it is possible to adjust the column layout, and select which columns should be included in the table.

6.7 Create Heat Map for Abundance Table

The Create Heat Map for Abundance Table tool simultaneously clusters samples and features (taxa), showing a two dimensional heat map of taxonomic abundances.

The following filtering and normalization is performed:

- 'log CPM' (Counts per Million) values are calculated for each feature. The CPM calculation uses the effective library sizes as calculated by the TMM normalization.
- Z-score normalization is performed across samples for each feature: the counts for each feature are mean centered, and scaled to unit variance.

For more detail about these steps, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=RNA_Seq_normalization.html.

6.7.1 Clustering of features and samples

Hierarchical clustering clusters taxa by the similarity of their taxonomic profiles over the set of samples, and samples by the similarity of taxonomic composition over the set of features (taxa).

Each clustering has a tree structure that is generated as follows:

1. Letting each taxa 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 times until there is only one cluster left (which contains all the taxa or samples).

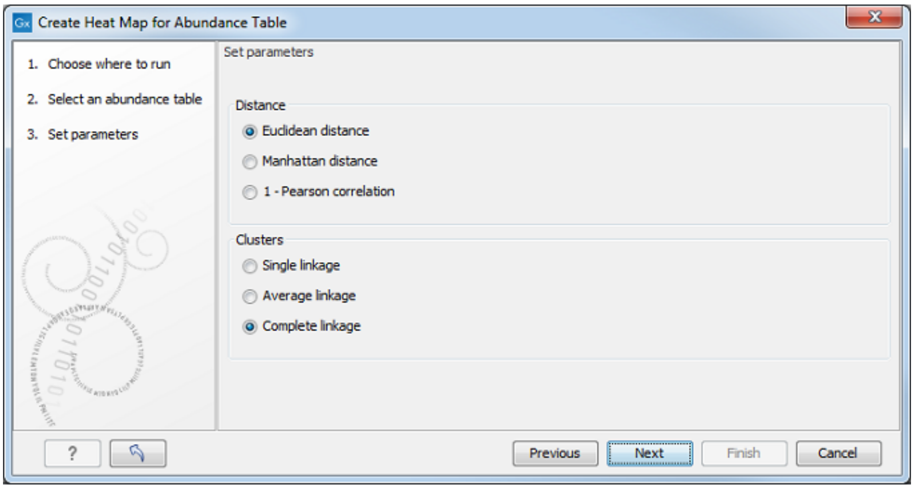
In the resulting tree, the length of branches reflect the distance between clusters.

To create a heat map:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **Create Heat Map for Abundance Table** ( )

Select an abundance table with two or more samples as input (e.g., a multi-sample OTU or merged abundance table) and click **Next**.

Specify a distance measure and a cluster linkage (figure 6.13). The distance measure is used to specify how distances between two taxa or samples should be calculated. The cluster linkage specifies how the distance between two clusters, each consisting of a number of taxa or samples, should be calculated. Learn more about how distances and clusters are calculated at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Clustering_features_samples.html.



Create Heat Map for Abundance Table

1. Choose where to run
2. Select an abundance table
3. Set parameters

Set parameters

Distance
- ● Euclidean distance
- ○ Manhattan distance
- ○ 1 - Pearson correlation

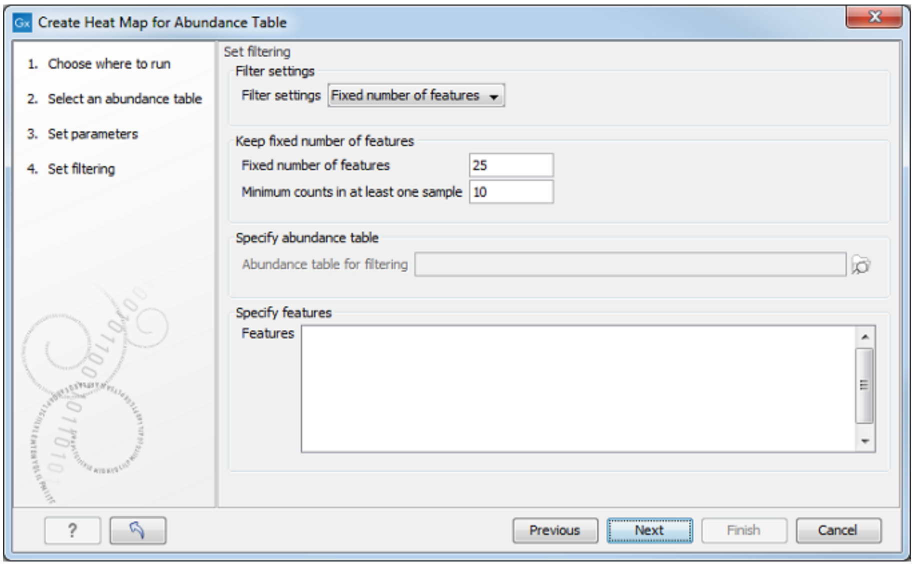
Clusters
- ○ Single linkage
- ○ Average linkage
- ● Complete linkage

Figure 6.13: *Select an abundance table.*

After having selected the distance measure, set up the feature filtering options (figure 6.14).

Genomes usually contain too many features to allow for a meaningful visualization. Clustering hundreds of thousands of features is also very time consuming. We therefore recommend to reduce the number of features before clustering, using the filter options available:

- 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.
- Abundance table: Specify a subset of an abundance table in case you only want to display the heat map for that particular subset. Note that creating the heat map from the subset


Create Heat Map for Abundance Table

1. Choose where to run
2. Select an abundance table
3. Set parameters
4. Set filtering

Set filtering

Filter settings

Filter settings: Fixed number of features

Keep fixed number of features

Fixed number of features: 25

Minimum counts in at least one sample: 10

Specify abundance table

Abundance table for filtering

Specify features

Features

Figure 6.14: *Set filtering options.*

abundance table directly can not ensure proper normalization of the data, and it is therefore recommended to use the original abundance table as input and filter using this option.

- Specify features: Keeps a set of features, as specified by plain text, i.e., a list of feature names. Any white-space characters, as well as "," and ";" are accepted as separators.

6.7.2 The heat map view

The tool generates a heat map showing the abundance of each feature in each sample and showing the sample clustering and/or feature clustering as a binary tree over the samples and features, respectively (figure 6.15).

Each column corresponds to one sample, and each row corresponds to a taxon. Samples and features are hierarchically clustered. Available sample metadata is added as an overlay.

6.7.3 Create heat map for specific taxonomic level

To create a heat map with a particular taxonomic level, you first need to create an aggregated version of your abundance table:

- Open the abundance table.
- In the **Data** section of the **Side panel**, select the desired **Aggregate feature** value, for example *Genus*.
- Select the desired features from the aggregated table, or use Ctrl+A (⌘ +A on Mac) to select all.
- Click on **Create Abundance Subtable** at the bottom of the viewing area. This will create an abundance subtable from the selection.
- Save the subtable and use it as input for **Create Heat Map for Abundance Table**.

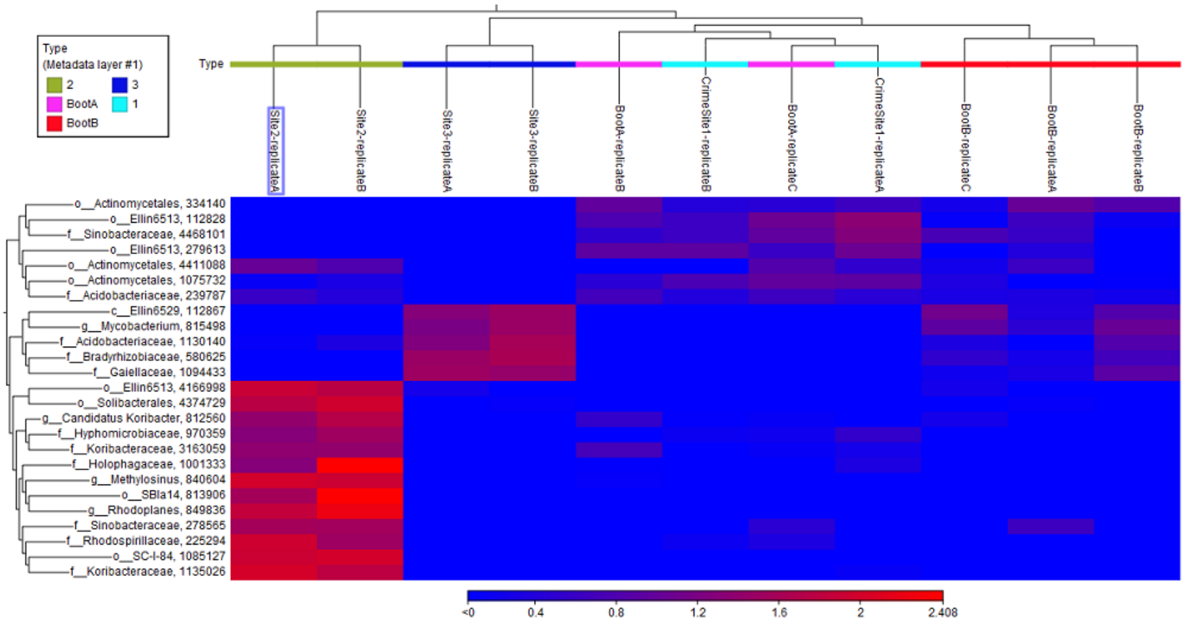


Figure 6.15: *Heat map.*

The resulting heat map will be grouped based on the selected taxonomic level.

6.8 Add Metadata to Abundance Table

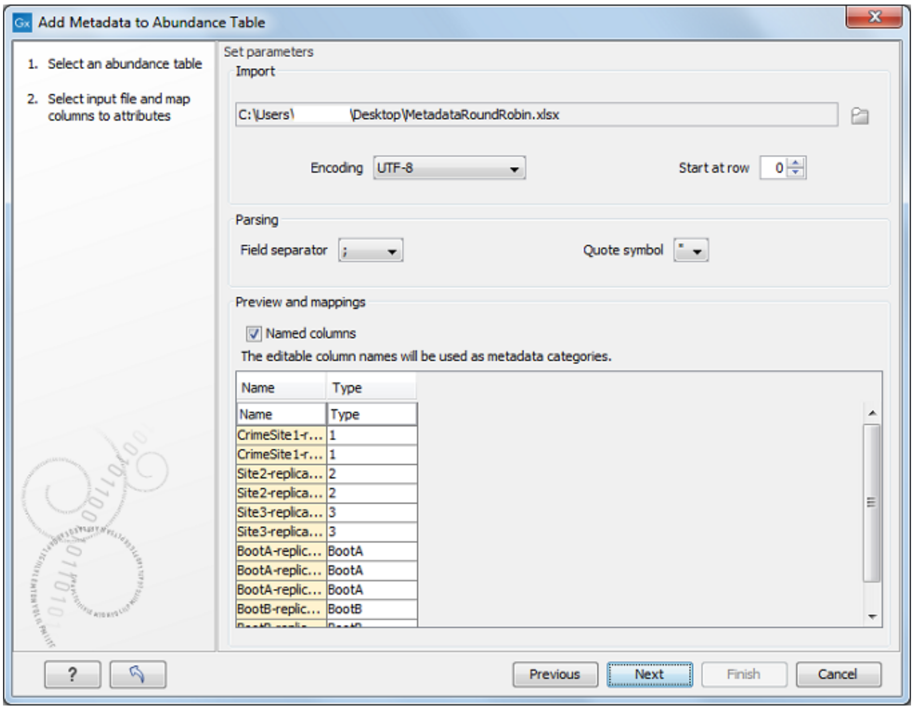
It is useful to have abundance tables decorated with sample metadata. This can be done by importing metadata and associating it with the reads before generating an abundance table. To learn more about how to create a metadata table, how to import a metadata table, or how to associate data elements with metadata, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Metadata.html.

If you did not have metadata associated with your reads prior to creating an abundance table, use **Add Metadata to Abundance Table** to add it retrospectively.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Abundance Analysis** ( ) | **Add Metadata to Abundance Table** ( )

Choose an abundance table as input. In the next wizard window you can select a file describing the metadata on your local computer (figure 6.16).



Add Metadata to Abundance Table

1. Select an abundance table
2. Select input file and map columns to attributes

Set parameters

Import

C:\Users\ \Desktop\MetadataRoundRobin.xlsx

Encoding UTF-8

Start at row 0

Parsing

Field separator ;

Quote symbol "

Preview and mappings

☑ Named columns

The editable column names will be used as metadata categories.

Name	Type
Name	Type
CrimeSite1-r...	1
CrimeSite1-r...	1
Site2-replica...	2
Site2-replica...	2
Site3-replica...	3
Site3-replica...	3
BootA-replic...	BootA
BootA-replic...	BootA
BootA-replic...	BootA
BootB-replic...	BootB

Figure 6.16: *Setting up metadata parameters.*

The metadata should be formatted in a tabular file format (*.xls, *.xlsx). The first row of the table should contain column headers. There should be one column called "Name" and the entries in this column should match the names of the reads selected for creating the abundance table. This column is used to match row in the table with samples present in the abundance table, so if the names do not match you will not be able to aggregate your data at all. There can be as many other columns as needed, and these information can be used as grouping variables to improve visualization of the results or to perform additional statistical analyses. If you wish to ignore a column without deleting it from your file, simply delete the text in the header row.

Note that when importing an Excel file, formulas will be imported as the formula text and not as the result of the calculation. If you utilize formulas in the metadata file you want to import, you have to flatten the file before importing. This can be done in a number of ways, for instance by exporting to a CSV file (and then importing that instead), or copying and using "Paste Special" in Excel: Start by selecting everything, copy the selection to the clipboard and then execute "Paste Special". On Windows "Paste Special" can be executed by holding Ctrl and Alt and then pressing V. On a Mac "Paste Special" can be executed by holding Ctrl and ⌘ and pressing V. Once the "Paste Special" dialog appears, select "Values" under "Paste" and finally click OK.

Part IV

Typing and Epidemiology

Chapter 7

Introduction to Typing and Epidemiology

Next generation sequencing (NGS) data from whole pathogen genomes is frequently used for enhanced surveillance and outbreak detection of common pathogens. CLC Microbial Genomics Module introduces functionality for molecular typing and epidemiological analysis of bacterial isolates. The module enables the user to perform a range of analyses, or to take advantage of template workflows for routine surveillance or outbreak analysis of a specific pathogen (figure 7.1).

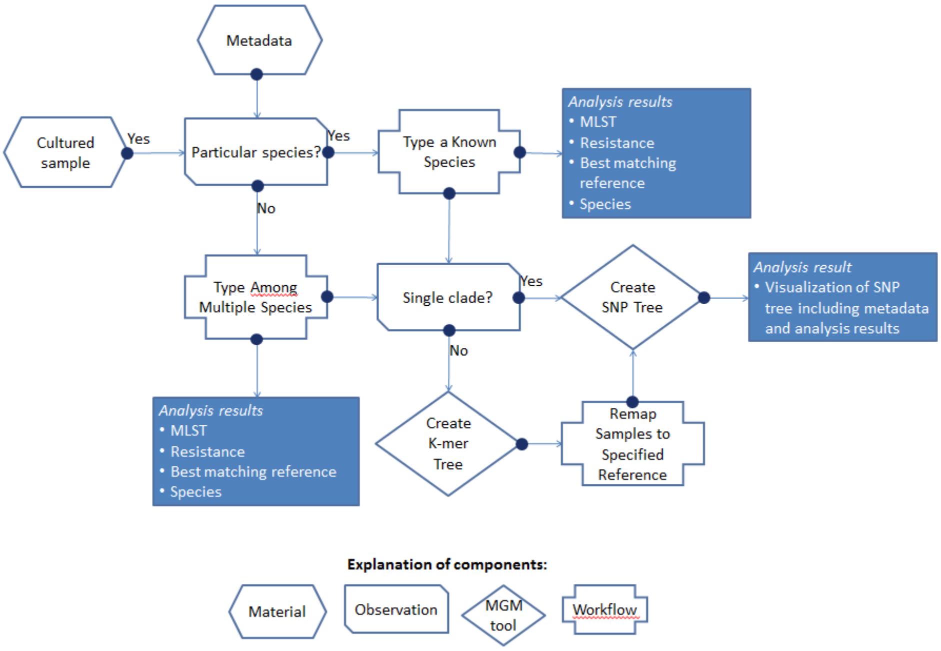


Figure 7.1: *Typing of cultured microbial samples shown as process diagram.*

The typing and epidemiology features include streamlined tools for NGS-based Multilocus Sequence Typing (MLST) and resistance typing, as well as fast detection of genus and species. It also includes tools for phylogenetic tree reconstruction based on single nucleotide polymorphisms (SNPs) or inference of K-mer trees from NGS reads or genomes. A new table format, acting as a database, collects typing results and associates these with metadata such as sample information, geographic origin, treatment outcome, etc. Results generated using NGS-MLST and resistance typing can hereby be associated with the original sample metadata. Users can filter on analysis results and metadata and then select relevant subsets of samples for downstream analysis. Results and metadata available during tree generation can also be used to explore and decorate this epidemiologically relevant information on the phylogenetic tree.

The Typing and Epidemiology tools are described in the following chapters.

Template workflows for typing and epidemiology analysis are available at:

> **Toolbox** | **Template Workflows** ( ) | **Microbial Workflows** ( ) | **Typing and Epidemiology** ( )

For more information, see section 2.3.

Chapter 8

Find the best matching reference

8.1 Find Best Matches using K-mer Spectra

The Find Best Matches using K-mer Spectra tool is inspired by Hasman et al., 2013 and Larsen et al., 2014 and enables identification of the best matching reference among a specified reference sequence list.

Template workflows for typing and epidemiology analysis are available at:

> **Toolbox** | **Template Workflows** () | **Microbial Workflows** () | **Typing and Epidemiology** ()

For more information, see section 2.3.

To identify best matching bacterial genome reference, go to:

> **Toolbox** | **Microbial Genomics Module** () | **Typing and Epidemiology** () | **Find Best Matches using K-mer Spectra** ()

Select the sequences you want want to find a best match sequence for (figure 8.1).

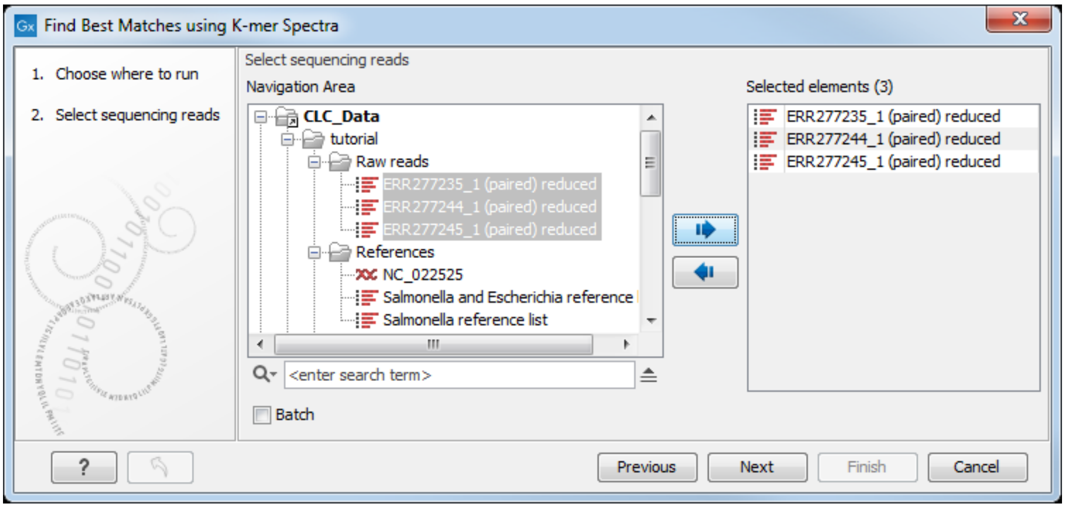
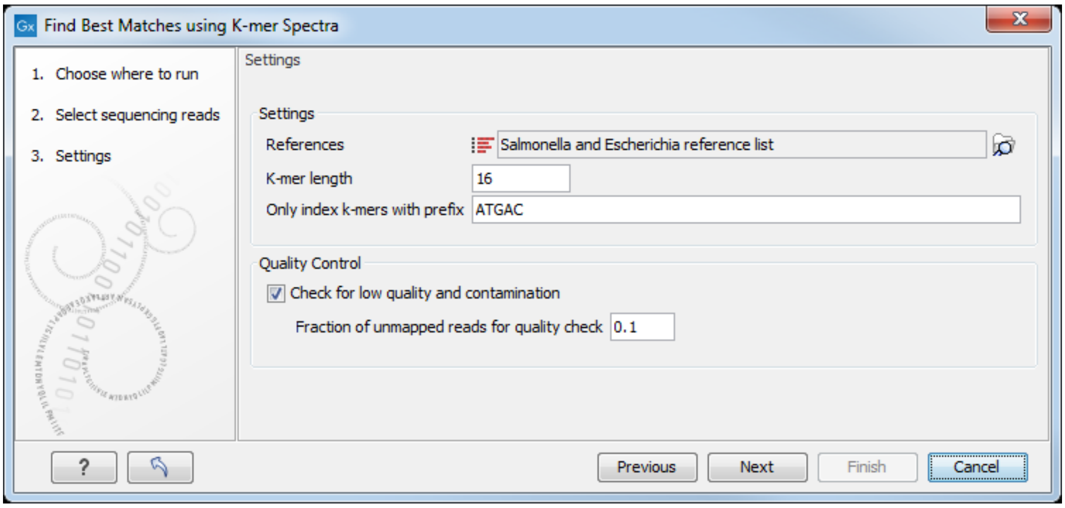

Find Best Matches using K-mer Spectra
1. Choose where to run
2. Select sequencing reads
Select sequencing reads
Navigation Area
CLC_Data
tutorial
Raw reads
ERR277235_1 (paired) reduced
ERR277244_1 (paired) reduced
ERR277245_1 (paired) reduced
References
NC_022525
Salmonella and Escherichia reference
Salmonella reference list
<enter search term>
Batch
Selected elements (3)
ERR277235_1 (paired) reduced
ERR277244_1 (paired) reduced
ERR277245_1 (paired) reduced
?
Previous
Next
Finish
Cancel

Figure 8.1: *To identify best matching reference, specification of read file is the first step.*

Select then a reference database, and specify the following settings (figure 8.2).



Find Best Matches using K-mer Spectra

1. Choose where to run
2. Select sequencing reads
3. Settings

Settings

Settings

References: Salmonella and Escherichia reference list

K-mer length: 16

Only index k-mers with prefix: ATGAC

Quality Control

☑ Check for low quality and contamination

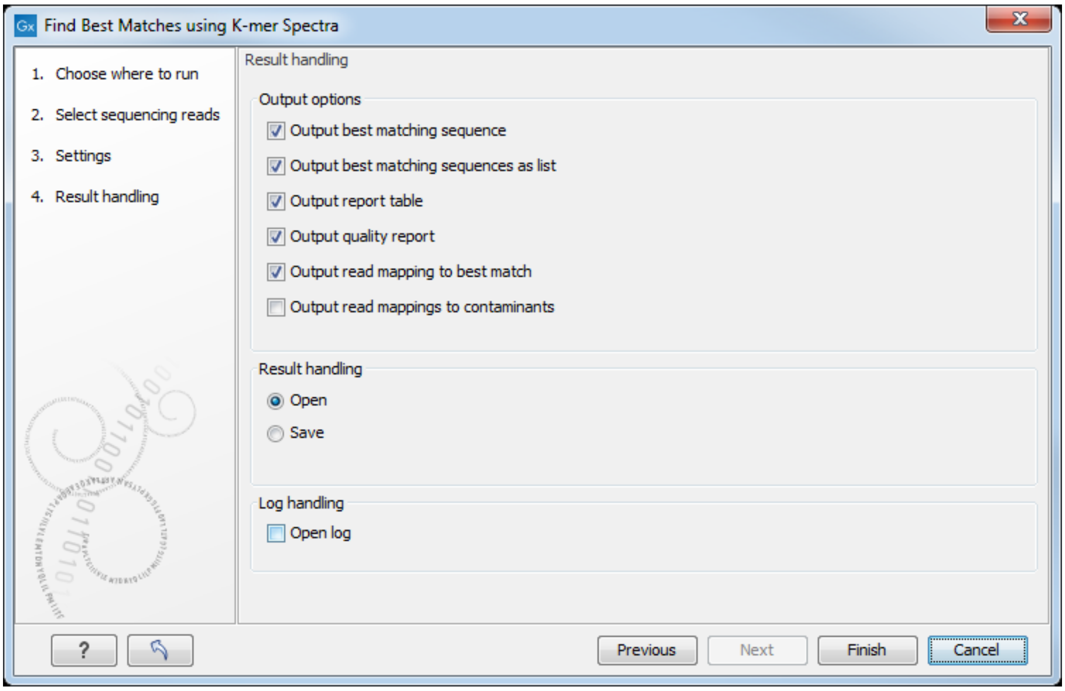
Fraction of unmapped reads for quality check: 0.1

Figure 8.2: *Specify reference list to search across.*

- **References** may be a single- or multiple list(s) of sequences. Sequences with identical entries in the Assembly ID and Latin Name columns are considered as one reference, see section 21. It is for example possible to use the full NCBI's bacterial genomes database, or subset(s) of it.
- **K-mer length** is the fixed number (k) of DNA bases to search across.
- **Only index k-mers with prefix** allows specification of the initial bases of the k-mer sequence to limit the search space.
- **Check for low quality and contamination** will perform a quality check of the input data and identify potential contaminations.
- **Fraction of unmapped reads for quality check** defines the contamination tolerance as the fraction of the total number of reads not mapping to the best reference.

In the last wizard window, the tool provides the following output options (figure 8.3).

- **Output Best Matching Sequence** is the best matching genome within the provided reference sequence list(s).
- **Output Best Matching Sequences as a List** includes the best matching genomes ordered with the best matching reference sequence first. The list is capped at 100 entries. Content is the same as in the Output Report Table.
- **Output Report Table** represents the best matching sequence. It lists all significantly matching references including various statistical values (as described in Hasman et al., 2013 and Larsen et al., 2014). The list is capped at 100 entries and the column headers are defined as such:
  - **Score** Numbers of k-mers from the database seen in the reads.
  - **Expected** The expected value, i.e., what score should been for the Z-score to be 0 and thus the P-value to be 1.
  - **Z** Calculated Z-score.



Find Best Matches using K-mer Spectra

1. Choose where to run
2. Select sequencing reads
3. Settings
4. Result handling

Result handling

Output options

☑ Output best matching sequence

☑ Output best matching sequences as list

☑ Output report table

☑ Output quality report

☑ Output read mapping to best match

□ Output read mappings to contaminants

Result handling

● Open

○ Save

Log handling

□ Open log

Figure 8.3: *Choose your output option before saving your results.*

  - **P** Z-score translated to two-sided P-value.
  - **P, corrected** P-value with Bonferroni correction.

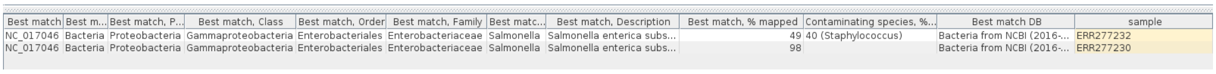
- **Output Quality Report** gives a report with some statistics on possible contamination and coverage reports for the read mappings. This option is available if the option **Check for low quality and contamination** was selected in the first wizard window. This report contains the metadata:
  - **Best match, % mapped** Percent of reads mapping to the best matching reference.
  - **Contaminating species, % mapped (taxonomy info)** Percent of mapping reads and the most specific accessible taxonomy information for the most probable contaminant.
- **Output read mapping to best match** gives the mapping of the reads to the best matching reference. This option is available if the option **Check for low quality and contamination** was selected in the first wizard window.
- **Output read mapping to contaminants** if a contamination is detected, this generates the mapping of the reads (which do not map to the best reference) to the probable contaminants. This option is available if the option **Check for low quality and contamination** was selected in the first wizard window.

In cases where the tool stops with a warning that good references were not found, you should download a new set of references for the organisms of interest and re-run the workflow.

To add the obtained best match to a Result Metadata Table, see section 19.2.3.

Note that in rare instances, the lists of references found in the **Output Best Matching Sequences as a List** and **Output Quality Report** may differ. The reason is that the former list is compiled based on a "Winner takes all" based count of K-mers which attributes all uniquely found K-mers *only* to the reference with the highest Z-score,. The latter list however is produced by removing all reads mapping to the best matching reference and using the remaining reads as a basis for determining the next best match. Thus, in the second round the pool of K-mers has been altered, and some K-mers that determined the Z-score of the original second-best match may have been removed.

Once results from the Find Best Matches using K-mer Spectra tool are added to the Result Metadata Table, extra columns are present in the table, including the taxonomy of the best matching references. In addition, in case the quality control was activated, the table will include the percentage of reads mapping to the best reference and the most probable contaminating species (see figure 8.4).



Best match	Best m...	Best match, P...	Best match, Class	Best match, Order	Best match, Family	Best matc...	Best match, Description	Best match, % mapped	Contaminating species, %...	Best match DB	sample
NC_017046	Bacteria	Proteobacteria	Gammaproteobacteria	Enterobacteriales	Enterobacteriaceae	Salmonella	Salmonella enterica subs...	49	40 (Staphylococcus)	Bacteria from NCBI (2016-...	ERR277232
NC_017046	Bacteria	Proteobacteria	Gammaproteobacteria	Enterobacteriales	Enterobacteriaceae	Salmonella	Salmonella enterica subs...	98		Bacteria from NCBI (2016-...	ERR277230

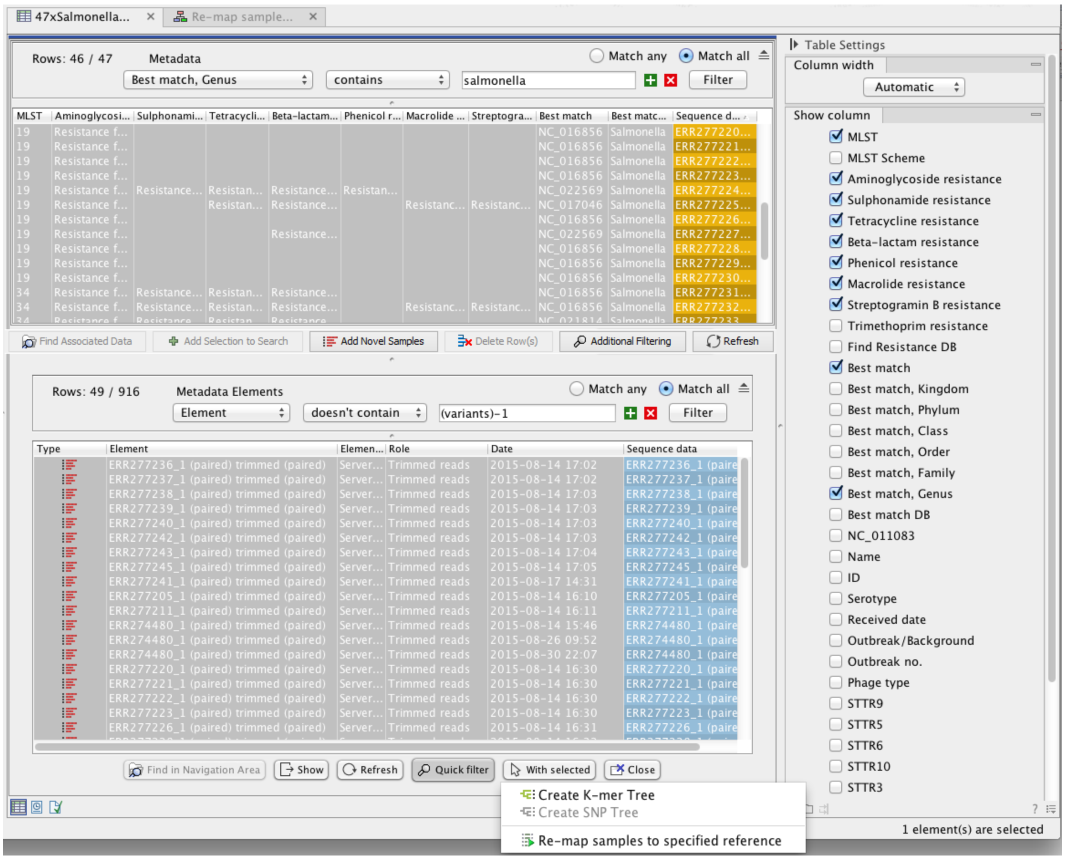
Figure 8.4: *Taxonomy of the best matching reference and quality information is shown in the Metadata Result Table.*

8.1.1 From samples best matches to a common reference for all

If several best matches are found across the samples, you probably want to find a common reference sequence to all (or a subset of) the samples. This can be done directly from your Metadata Result Table, by selecting the samples of interest and creating a K-mer Tree based on these samples (see figure 8.5).

1. **Select** in your Metadata result Table the samples to which a common best matching reference should be identified.
2. **Click** on the **Find Associated Data** ( ) button to find their associated Metadata Elements.
3. **Click** on the **Quick Filtering** ( ) button and select the option **Filter for K-mer Tree** to find Metadata Elements with the Role = Trimmed Reads.
4. **Select** the relevant Metadata Element files.
5. **Click** on the **With selected** ( ) button.
6. **Select** the **Create K-mer Tree** action.

Once you have selected the **Create K-mer Tree** action, you can follow the wizard as described in section 9.2. This section will also explain how to understand the tree and continue with subsequent analyses. In short, the common reference is chosen as the genome sharing the closest common ancestor with the clade of isolates under study in the k-mer tree.



Rows: 46 / 47 Metadata ○ Match any ● Match all

Best match, Genus | contains | salmonella | Filter

MLST	Aminoglycosi...	Sulphonami...	Tetracycli...	Beta-lactam...	Phenicol r...	Macrolide ...	Streptogra...	Best match	Best matc...	Sequence d...
19	Resistance f...							NC_016856	Salmonella	ERR277220...
19	Resistance f...							NC_016856	Salmonella	ERR277221...
19	Resistance f...							NC_016856	Salmonella	ERR277222...
19	Resistance f...							NC_016856	Salmonella	ERR277223...
19	Resistance f...	Resistance...	Resistan...	Resistance...	Resistan...			NC_022569	Salmonella	ERR277224...
19	Resistance f...		Resistan...	Resistance...		Resistanc...	Resistanc...	NC_017046	Salmonella	ERR277225...
19	Resistance f...							NC_016856	Salmonella	ERR277226...
19	Resistance f...			Resistance...				NC_022569	Salmonella	ERR277227...
19	Resistance f...							NC_016856	Salmonella	ERR277228...
19	Resistance f...							NC_016856	Salmonella	ERR277229...
19	Resistance f...							NC_016856	Salmonella	ERR277230...
34	Resistance f...	Resistance...	Resistan...	Resistance...				NC_016856	Salmonella	ERR277231...
34	Resistance f...	Resistance...	Resistan...	Resistance...		Resistanc...	Resistanc...	NC_016856	Salmonella	ERR277232...

Rows: 49 / 916 Metadata Elements ○ Match any ● Match all

Element | doesn't contain | (variants)-1 | Filter

Type	Element	Elemen...	Role	Date	Sequence data
	ERR277236_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:02	ERR277236_1 (paire
	ERR277237_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:02	ERR277237_1 (paire
	ERR277238_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:03	ERR277238_1 (paire
	ERR277239_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:03	ERR277239_1 (paire
	ERR277240_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:03	ERR277240_1 (paire
	ERR277242_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:03	ERR277242_1 (paire
	ERR277243_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:04	ERR277243_1 (paire
	ERR277245_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 17:05	ERR277245_1 (paire
	ERR277241_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-17 14:31	ERR277241_1 (paire
	ERR277205_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:10	ERR277205_1 (paire
	ERR277211_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:11	ERR277211_1 (paire
	ERR274480_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 15:46	ERR274480_1 (paire
	ERR274480_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-26 09:52	ERR274480_1 (paire
	ERR274480_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-30 22:07	ERR274480_1 (paire
	ERR277220_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:30	ERR277220_1 (paire
	ERR277221_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:30	ERR277221_1 (paire
	ERR277222_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:30	ERR277222_1 (paire
	ERR277223_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:30	ERR277223_1 (paire
	ERR277226_1 (paired) trimmed (paired)	Server...	Trimmed reads	2015-08-14 16:31	ERR277226_1 (paire

Column width: Automatic

Show column

- ☑ MLST
- ☐ MLST Scheme
- ☑ Aminoglycoside resistance
- ☑ Sulphonamide resistance
- ☑ Tetracycline resistance
- ☑ Beta-lactam resistance
- ☑ Phenicol resistance
- ☑ Macrolide resistance
- ☑ Streptogramin B resistance
- ☐ Trimethoprim resistance
- ☐ Find Resistance DB
- ☑ Best match
- ☐ Best match, Kingdom
- ☐ Best match, Phylum
- ☐ Best match, Class
- ☐ Best match, Order
- ☐ Best match, Family
- ☑ Best match, Genus
- ☐ Best match DB
- ☐ NC_011083
- ☐ Name
- ☐ ID
- ☐ Serotype
- ☐ Received date
- ☐ Outbreak/Background
- ☐ Outbreak no.
- ☐ Phage type
- ☐ STTR9
- ☐ STTR5
- ☐ STTR6
- ☐ STTR10
- ☐ STTR3

1 element(s) are selected

Figure 8.5: *Once samples are selected in the top window, it is easy to find the associated Metadata Element files, Quick Filter towards generation of K-mer tree and finally initiate creation of K-mer tree to be used for identification of common reference sequence. From the same view, it is also possible to run a customized version of the Map to Specified Reference workflow with the selected elements.*

8.2 Find Best References using Read Mapping

The Find Best References using Read Mapping tool maps reads to a reference sequence list to identify the best matching reference i.e., the references for which the input reads hold more evidence.

If a host genome is provided, reads that map better to the host are filtered to not have them count toward results.

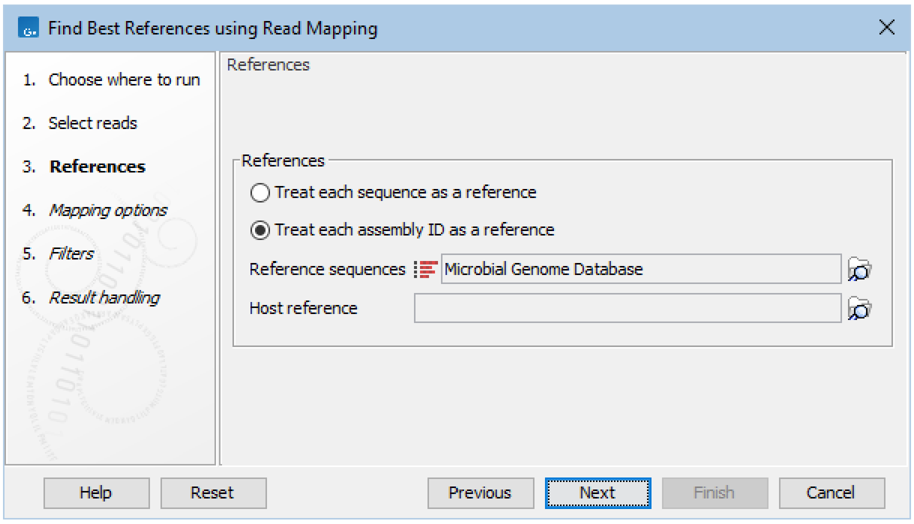
To start the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Typing and Epidemiology** ( ) | **Find Best References using Read Mapping** ( )

In the first dialog, select the sequences or sequence lists containing the sequencing reads, and click on **Next**.

In the **References** dialog, specify the following (figure 8.6):

- **Treat each sequence as a reference**. Each sequence makes up a separate reference.
- **Treat each assembly ID as a reference**. Sequences with the same assembly ID make up


Find Best References using Read Mapping

1. Choose where to run
2. Select reads
3. **References**
4. *Mapping options*
5. *Filters*
6. *Result handling*

References

References

○ Treat each sequence as a reference

● Treat each assembly ID as a reference

Reference sequences: Microbial Genome Database

Host reference:

Figure 8.6: *Select references.*

one reference and will be reported as such. This supports segmented references.

- **Reference sequence**. Select the reference sequence list.
  The tool is able to handle duplicate references. If same-name references have identical sequences, only one of these will be included in analysis. If same-name references have different sequences, they will be renamed to ensure unique names.

- **Host reference**. If relevant, provide a host reference to filter reads that map better to the host genome than to the reference sequences.

In the **Mapping options** dialog, specify settings for the read mapping (figure 8.7). The options are identical to those of the Map Reads to Reference tool and are described here: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters.html.

The **Filters** dialog holds the following options (figure 8.8):

- **Minimum count**. Minimum number of mapped reads required for a reference to be reported.

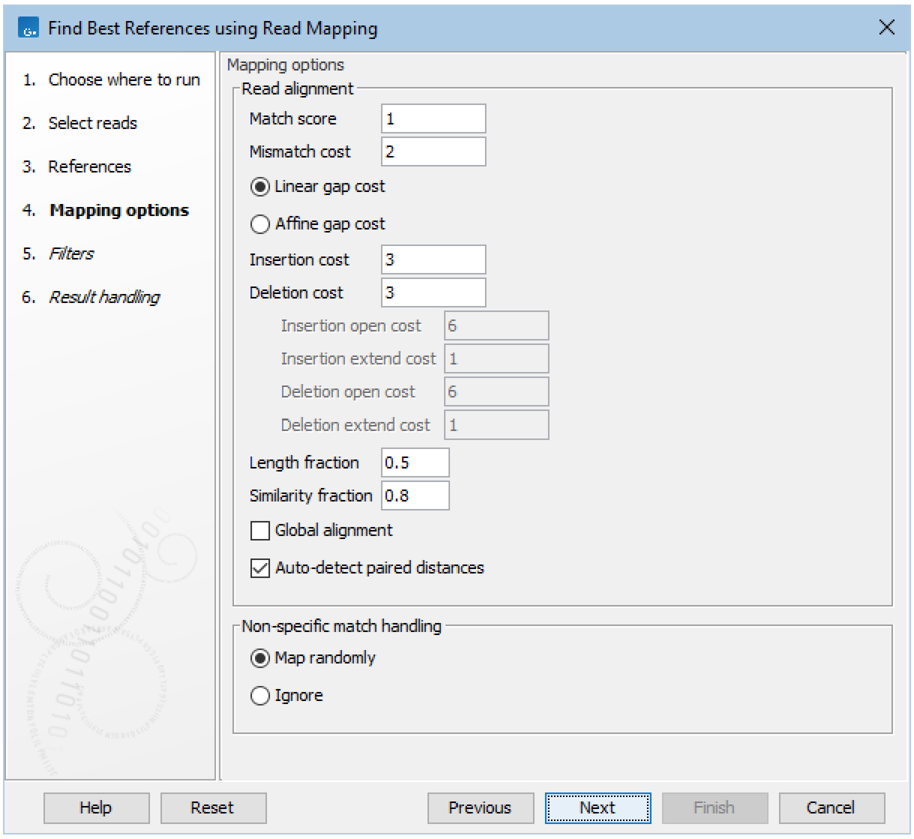
- **Minimum relative abundance**. Minimum relative abundance compared to most abundant reference required for a reference to be reported.

- **Minimum fraction of reference covered**. Minimum fraction of the reference sequence to be covered by at least one read for a reference to be reported.

- **Minimum average coverage**. Minimum average coverage for a reference to be reported. Average coverage: Number of nucleotides mapped to a reference divided by the reference length.

- **Maximum number of references to report**. The maximum number of references to report. References are ranked according to the number of mapped reads.

In the final step, specify the output:



Mapping options

Read alignment

Match score: 1

Mismatch cost: 2

● Linear gap cost

○ Affine gap cost

Insertion cost: 3

Deletion cost: 3

Insertion open cost: 6

Insertion extend cost: 1

Deletion open cost: 6

Deletion extend cost: 1

Length fraction: 0.5

Similarity fraction: 0.8

□ Global alignment

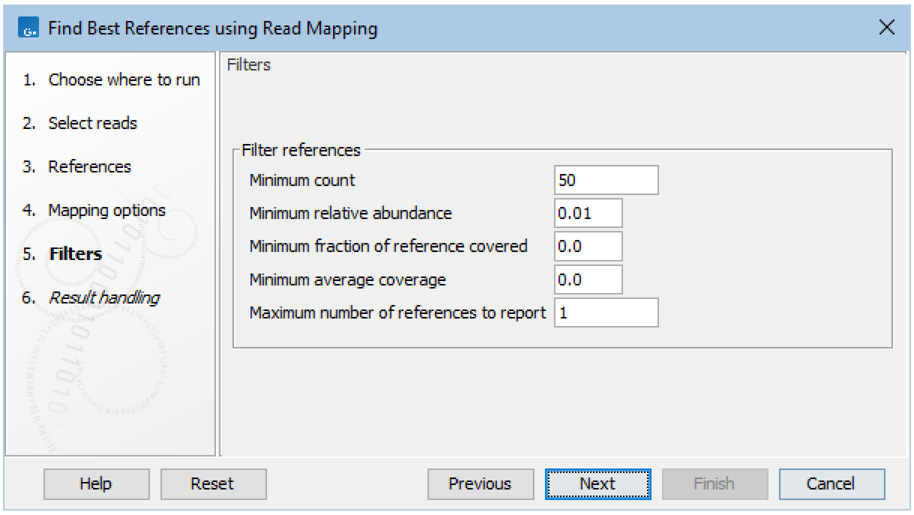
☑ Auto-detect paired distances

Non-specific match handling

● Map randomly

○ Ignore

Figure 8.7: *Select mapping options.*



Filters

Filter references

Minimum count: 50

Minimum relative abundance: 0.01

Minimum fraction of reference covered: 0.0

Minimum average coverage: 0.0

Maximum number of references to report: 1

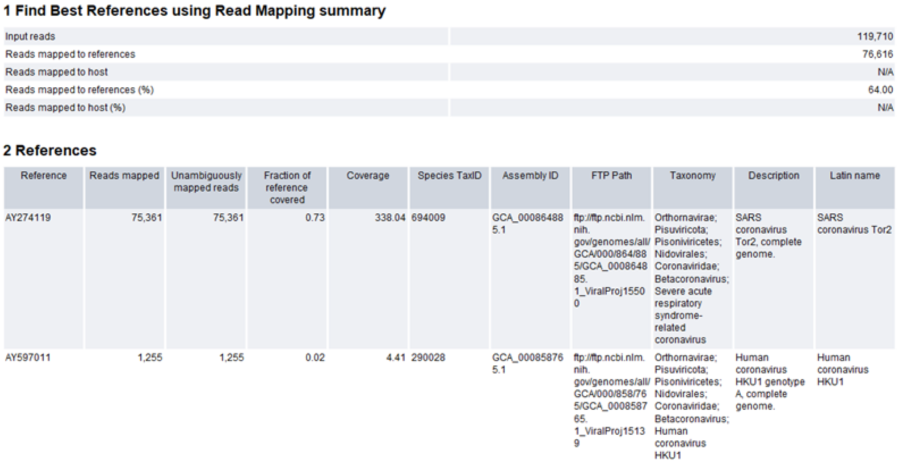
Figure 8.8: *Select filtering options.*

- **Create reference sequence list.** A sequence list with the identified best-match reference sequences.
- **Create reads track.** A track of reads mapped to the reference sequence(s).
- **Create reads track (host).** A track of reads mapped to the host reference.

- **Create report**. A summary report (section 8.2.1).

8.2.1 The Find Best References using Read Mapping Report

The Find Best References using Read Mapping report contains a summary of the read mapping results and a table of the identified best-match references (figure 8.9).



1 Find Best References using Read Mapping summary

Input reads	119,710
Reads mapped to references	76,616
Reads mapped to host	N/A
Reads mapped to references (%)	64.00
Reads mapped to host (%)	N/A

2 References

Reference	Reads mapped	Unambiguously mapped reads	Fraction of reference covered	Coverage	Species TaxID	Assembly ID	FTP Path	Taxonomy	Description	Latin name
AY274119	75,361	75,361	0.73	338.04	694009	GCA_000864885.1	ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/864/885/GCA_000864885.1_ViralProj15500	Orthornavirae; Pisuviricota; Pisoniviricetes; Nidovirales; Coronaviridae; Betacoronavirus; Severe acute respiratory syndrome-related coronavirus	SARS coronavirus Tor2, complete genome.	SARS coronavirus Tor2
AY597011	1,255	1,255	0.02	4.41	290028	GCA_000858765.1	ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/858/765/GCA_000858765.1_ViralProj15139	Orthornavirae; Pisuviricota; Pisoniviricetes; Nidovirales; Coronaviridae; Betacoronavirus; Human coronavirus HKU1	Human coronavirus HKU1 genotype A, complete genome.	Human coronavirus HKU1

Figure 8.9: *The report for Find Best References using Read Mapping.*

Chapter 9

Phylogenetic trees using SNPs or k-mers

9.1 Create SNP Tree

The **Create SNP Tree** tool is inspired by Kaas et al., 2014.

To generate a SNP tree, first map reads from the individual samples to a common reference and call variants. The corresponding tools are described at:

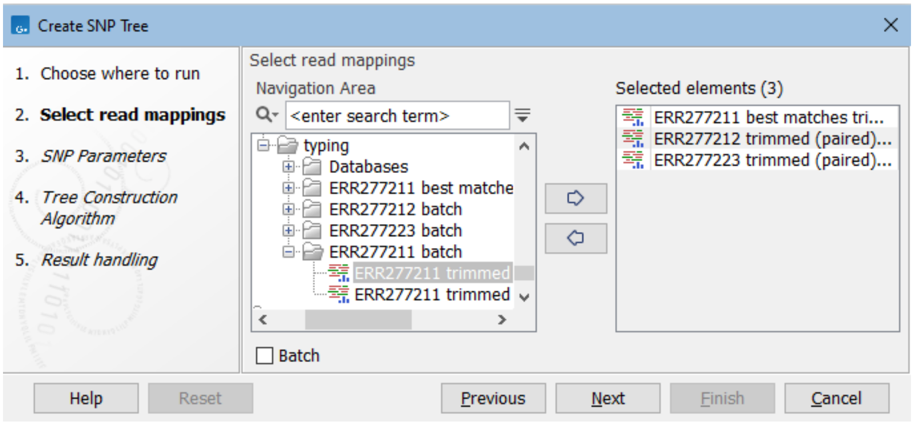
- Map Reads To Reference: `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters.html`

- Variant detection: `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Variant_detection.html`

To create a SNP tree, go to:

> **Toolbox** | **Microbial Genomics Module** () | **Typing and Epidemiology** () | **Create SNP Tree** ()

In the first dialog, select reads tracks or read mappings (figure 9.1).

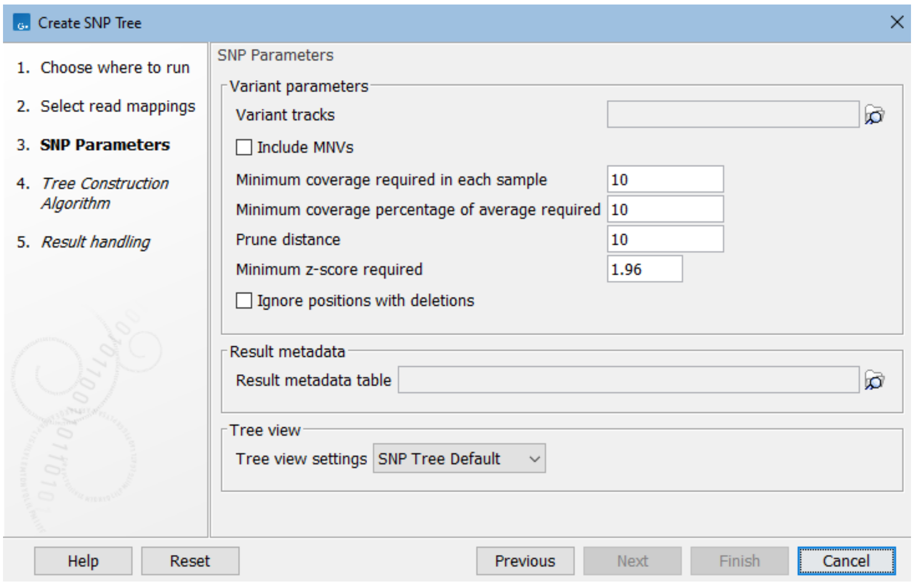


Selected elements (3)
ERR277211 best matches tri...
ERR277212 trimmed (paired)...
ERR277223 trimmed (paired)...

Figure 9.1: *Select read mappings to be included in the SNP tree analysis.*

Next, select **Variant parameters**. These determine which SNPs (single-nucleotide polymorphisms) and MNVs (multi-nucleotide variants) to consider for building the SNP tree:

- **Variant track**. Select variant tracks that correspond to the previously selected reads tracks or read mappings (figure 9.2). The variant tracks determine which positions to potentially include in the SNP tree.



Create SNP Tree

1. Choose where to run
2. Select read mappings
3. **SNP Parameters**
4. *Tree Construction Algorithm*
5. *Result handling*

SNP Parameters

Variant parameters

Variant tracks

□ Include MNVs

Minimum coverage required in each sample: 10

Minimum coverage percentage of average required: 10

Prune distance: 10

Minimum z-score required: 1.96

□ Ignore positions with deletions

Result metadata

Result metadata table

Tree view

Tree view settings: SNP Tree Default

Figure 9.2: *Select variant tracks and specify relevant parameters before generation of a SNP tree.*

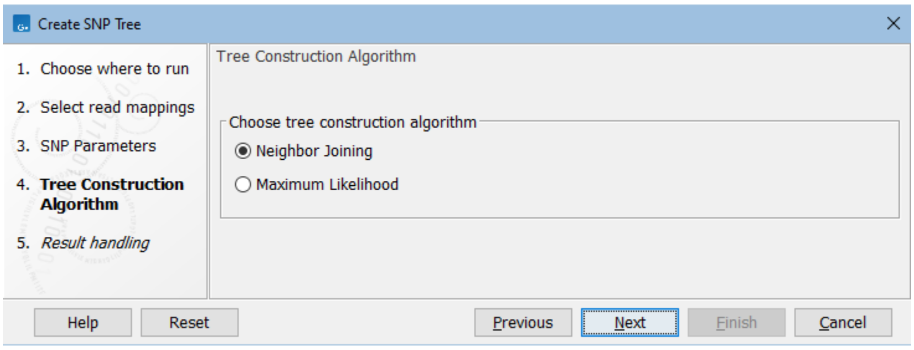
- **Include MNVs**. Check this option to include MNVs along with SNPs when building the SNP tree.
- **Minimum coverage required in each sample**. Positions are filtered if at least one sample has coverage below the specified threshold.
- **Minimum coverage percentage of average required**. Positions are filtered if the coverage of at least one sample falls below the specified percentage of the average coverage of that sample.
- **Prune distance**. Minimum number of nucleotides between unfiltered positions. If a position is within this distance of a previously used position it will be filtered.
- **Minimum z-score required**. Defining x as the number of the most prevalent nucleotide at a position and y as the coverage subtracting x, the z-score is calculated as $z = \frac{x-y}{\sqrt{x+y}}$. If the calculated z-score for a given position is less than the specified minimum value the position is filtered.
- **Ignore positions with deletions**. Check this option to ignore SNP positions where at least one sample has a deletion at the given position.

The initial list of SNP positions is reduced based on the above filters. Of the remaining, only variants with relative frequency above 50% (haploid organisms) will be considered. Information about reference and alleles is deduced from the read mappings.

Select the **Result metadata** table with metadata relevant for your samples. This will allow you to decorate the resulting SNP tree with metadata information, see section 9.1.2.

Select **Tree view settings.** (None, K-mer Tree Default, or SNP Tree Default) or your own custom tree setting. Read more on tree settings in general at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.

In the next dialog, select the tree construction algorithm (figure 9.3).



Create SNP Tree

1. Choose where to run
2. Select read mappings
3. SNP Parameters
4. **Tree Construction Algorithm**
5. *Result handling*

Tree Construction Algorithm

Choose tree construction algorithm

● Neighbor Joining

○ Maximum Likelihood

Figure 9.3: *Choose the tree construction algorithm.*

- **Neighbor Joining**. Branch length is based on the distance between samples computed as *Positions used where the consensus sequence is different / Total positions*. If you move from one sample to another in the tree, the sum of the lengths of the branches traversed equals the distance between those samples.
- **Maximum Likelihood**. Creates an initial tree using the Neighbor Joining method and subsequently calculates the most likely phylogenetic tree under the given evolutionary model. Parameters for this method are defined in the next dialog (see figure 9.4).

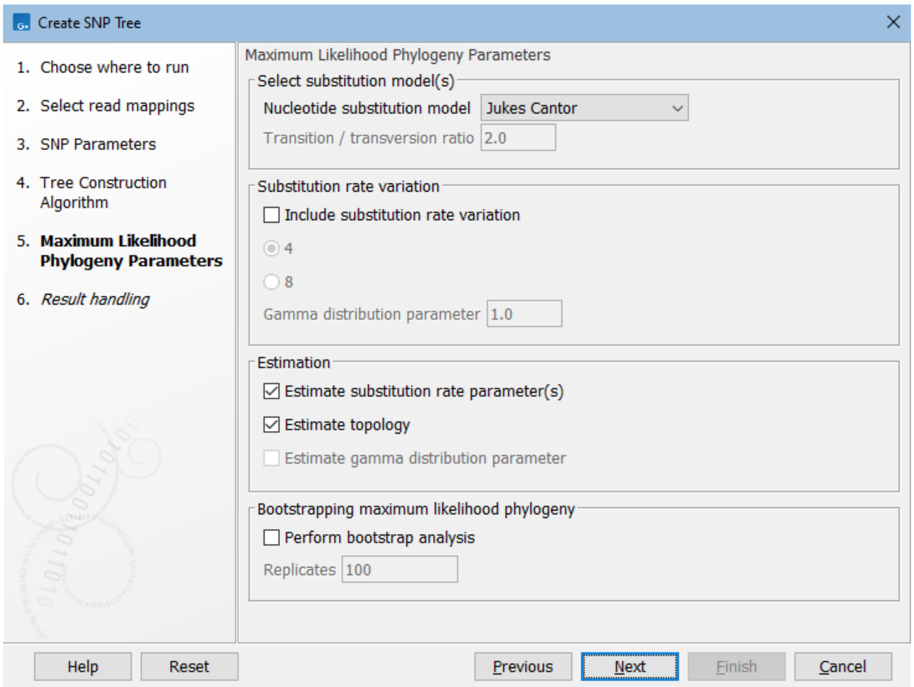
If you selected *Maximum Likelihood*, the next dialog covers parameters for this algorithm (see figure 9.4). The parameters are described here: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Maximum_Likelihood_Phylogeny.html.

In the **Result handling** dialog, specify the output (figure 9.5).

In addition to the SNP tree, the following are available:

- **Create report**. A SNP report with summary of input and results.
- **Create SNP alignment**. Outputs the alignment of concatenated SNPs that is produced as a first step in the algorithm.

  The alignment can be used as input for the Model Testing tool that serves to identify which evolutionary model suits the data best. Based on this, you may want to rerun the Create SNP Tree tool with adjusted settings. The Model Testing tool is described at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Model_Testing.html.
- **Create SNP matrix**. A matrix containing the number of SNP differences between all pairs of samples.



Maximum Likelihood Phylogeny Parameters

Select substitution model(s)

Nucleotide substitution model: Jukes Cantor

Transition / transversion ratio: 2.0

Substitution rate variation

□ Include substitution rate variation

● 4

○ 8

Gamma distribution parameter: 1.0

Estimation

☑ Estimate substitution rate parameter(s)

☑ Estimate topology

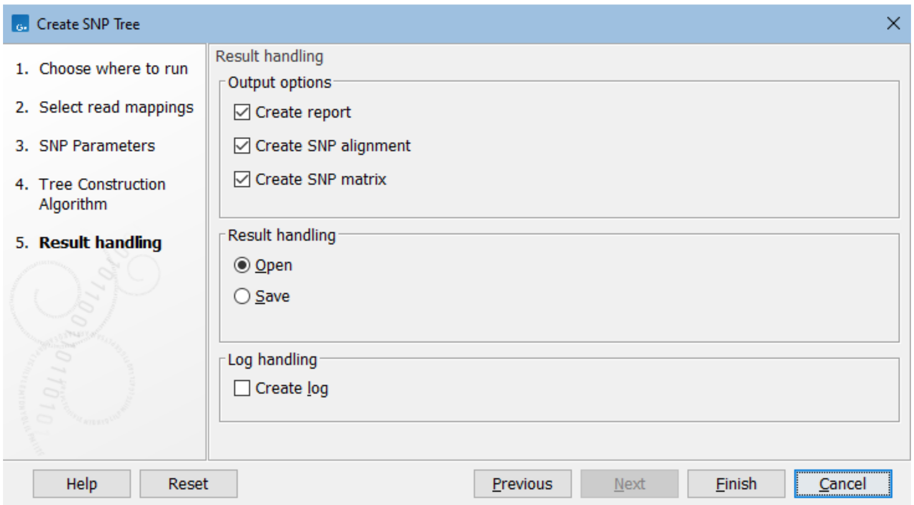
□ Estimate gamma distribution parameter

Bootstrapping maximum likelihood phylogeny

□ Perform bootstrap analysis

Replicates: 100

Figure 9.4: *Set parameters for maximum likeihood estimation.*



Result handling

Output options

☑ Create report

☑ Create SNP alignment

☑ Create SNP matrix

Result handling

● Open

○ Save

Log handling

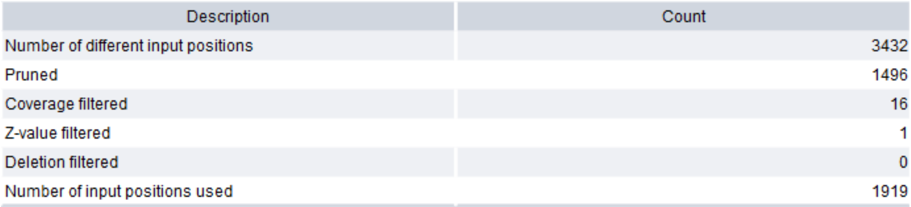
□ Create log

Figure 9.5: *Create SNP Tree output options.*

9.1.1 SNP tree report

The SNP tree report summarizes the result of the applied filtering.

- **Filter Status** (figure 9.6)
  - **Number of different input positions.** Unique SNPs (and MNVs, if selected) in the input variant tracks, pre-filtering.
  - **Pruned**. Positions filtered due to the prune distance threshold.
  - **Coverage filtered**. Positions filtered based on the two coverage filters.



Description	Count
Number of different input positions	3432
Pruned	1496
Coverage filtered	16
Z-value filtered	1
Deletion filtered	0
Number of input positions used	1919

Figure 9.6: *SNP tree report - Filter Status section.*

  - **Z-value filtered**. Positions filtered based on the minimum Z-value threshold.
  - **Deletion filtered**. Positions deleted as at least one sample has a deletion at this position.
  - **Number of input positions used**. Positions that passed all filtering steps and were included in the SNP tree.

- **Ignored positions attributed to read mappings**. Information on the number of positions filtered in the individual read mappings (figure 9.7).
  - **Read mapping**. The name of the read mapping.
  - **Filtered, total**. The number of SNPs from this read mapping that were filtered.
  - **Filtered, only by this**. The number of SNPs that were unique to this read mapping, and were filtered.

    If one or a few samples have a substantially higher number of filtered positions compared to the rest, one might consider rerunning the tree without these to improve the tree resolution.

9.1.2 SNP tree

The SNP tree can be vizualized in Tree view (), Table view (), and SNP Tree Variants view ().

Tree view ()

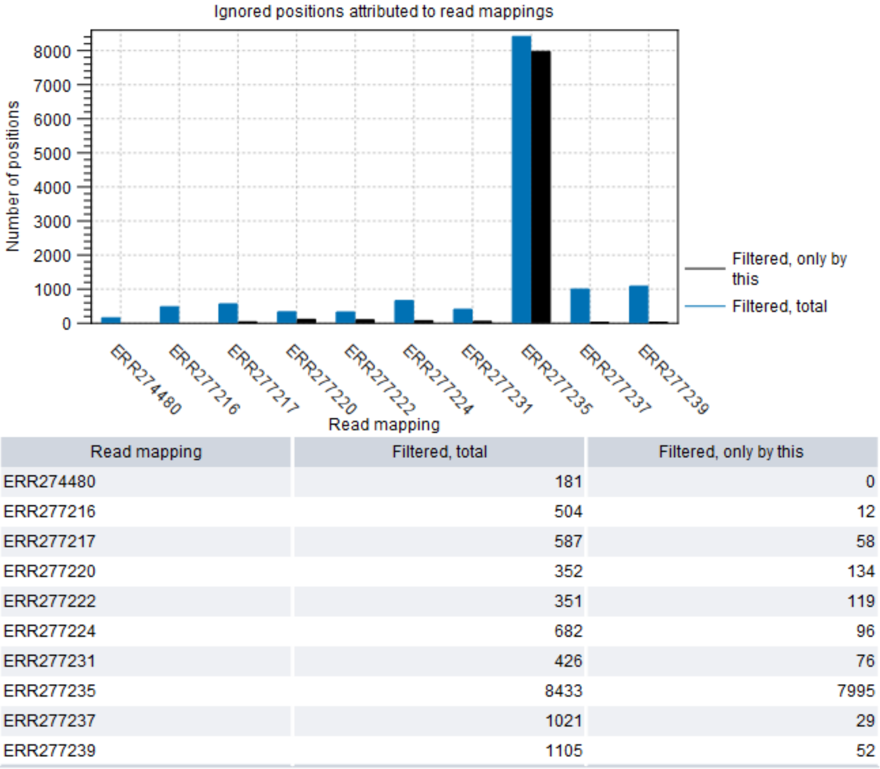
The SNP tree layout, node and label settings is adjusted from the Tree Settings Side Panel found in the left side of the view area. For details about the settings, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.

If a Result Metadata Table was provided as input, it is possible to decorate the SNP tree with one or more metadata layers from the Side Panel section **Metadata**, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Visualizing_metadata.html. This enables visual identification of potential correlation between samples, metadata layers, and tree typology (figure 9.8).

SNP Tree Variants view ()

With the SNP Tree Variants view it is possible to inspect the variants relating to a given internal node in the SNP tree. To populate the table with the SNPs of interest, first select the internal node of interest in the Tree view and then go to the **SNP Tree Variants** () view (figure 9.9).

The table lists the following columns:



Read mapping	Filtered, total	Filtered, only by this
ERR274480	181	0
ERR277216	504	12
ERR277217	587	58
ERR277220	352	134
ERR277222	351	119
ERR277224	682	96
ERR277231	426	76
ERR277235	8433	7995
ERR277237	1021	29
ERR277239	1105	52

Figure 9.7: *Visualization of the filter effect across data used for generation of SNP tree. One sample shows a higher number of filtered SNPs and could potentially be omitted from a new SNP tree.*

- **Position**. The position of the SNP on the reference chromosome
- **Chromosome**. The name of the reference chromosome.
- **All agree**. Yes/No indicates whether all samples belonging to the internal node have the same variant allele.

With the Side Panel section **SNP information** you can choose to have SNP information presented *As summary* or *By sample*:

- **As summary**. The table contains one column per subtree with a summary of alleles in the subtree samples e.g., "A (3), G (1)" (figure 9.9 - top image).
- **By sample**. The table contains one column per sample with sample-specific alleles (figure 9.9 - bottom image).

9.1.3 SNP Matrix

The SNP Matrix contains the pairwise number of SNP differences between all pairs of samples (see figure 9.10).

Use the Side Panel setting **Comparison gradient** to get an overview of which samples are closely related. Drag the arrows to change the minimum and maximum values of the scale, or click the gradient to access the gradient configuration dialog. Use the Lower threshold field to type in a

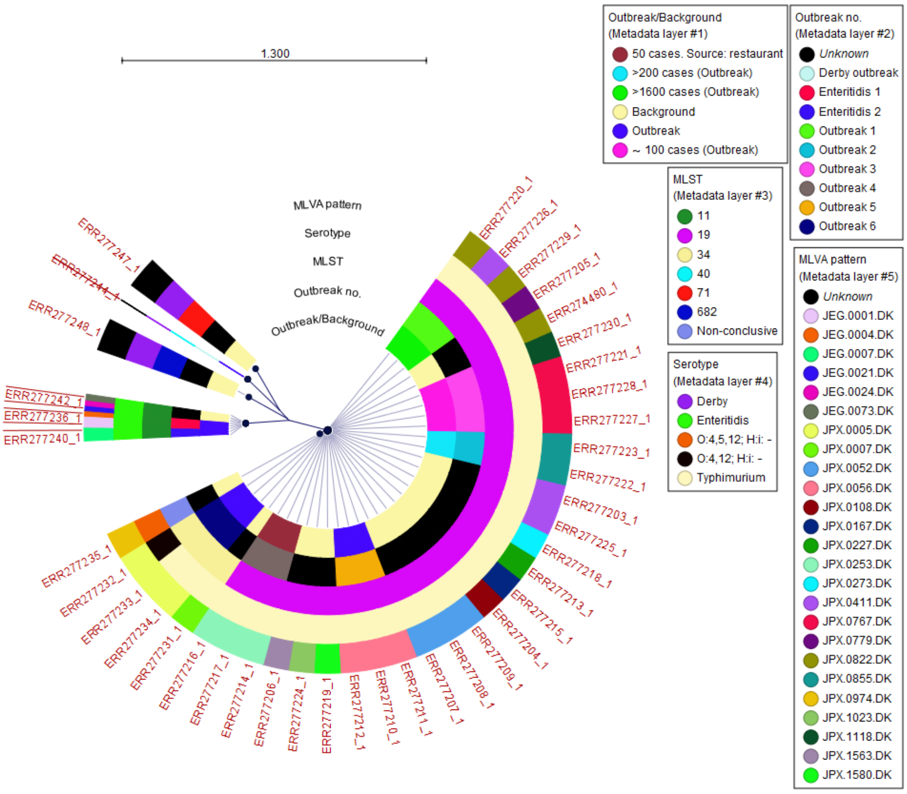
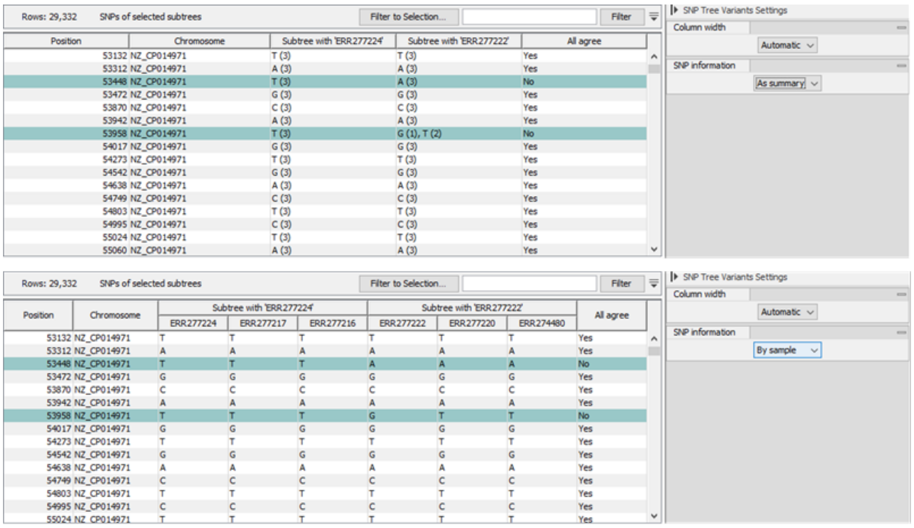


Figure 9.8: *Visualization of SNP tree including selected metadata and analysis result metadata.*



Rows: 29,332 SNPs of selected subtrees

Position	Chromosome	Subtree with 'ERR277224'	Subtree with 'ERR277222'	All agree
53132	NZ_CP014971	T (3)	T (3)	Yes
53312	NZ_CP014971	A (3)	A (3)	Yes
53448	NZ_CP014971	T (3)	A (3)	No
53472	NZ_CP014971	G (3)	G (3)	Yes
53870	NZ_CP014971	C (3)	C (3)	Yes
53942	NZ_CP014971	A (3)	A (3)	Yes
53958	NZ_CP014971	T (3)	G (1), T (2)	No
54017	NZ_CP014971	G (3)	G (3)	Yes
54273	NZ_CP014971	T (3)	T (3)	Yes
54542	NZ_CP014971	G (3)	G (3)	Yes
54638	NZ_CP014971	A (3)	A (3)	Yes
54749	NZ_CP014971	C (3)	C (3)	Yes
54803	NZ_CP014971	T (3)	T (3)	Yes
54995	NZ_CP014971	C (3)	C (3)	Yes
55024	NZ_CP014971	T (3)	T (3)	Yes
55060	NZ_CP014971	A (3)	A (3)	Yes

Column width: Automatic

SNP information: As summary

Rows: 29,332 SNPs of selected subtrees

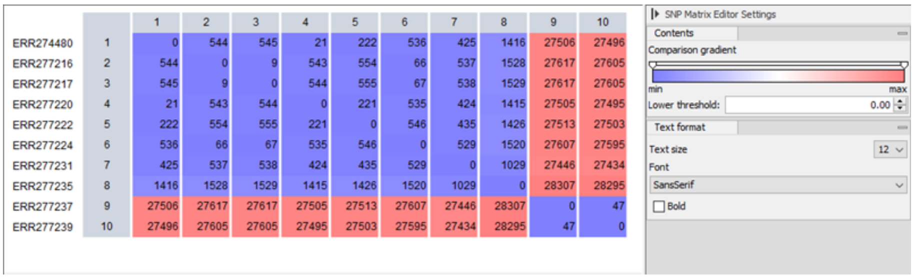
Position	Chromosome	Subtree with 'ERR277224'			Subtree with 'ERR277222'			All agree
		ERR277224	ERR277217	ERR277216	ERR277222	ERR277220	ERR274480	
53132	NZ_CP014971	T	T	T	T	T	T	Yes
53312	NZ_CP014971	A	A	A	A	A	A	Yes
53448	NZ_CP014971	T	T	T	A	A	A	No
53472	NZ_CP014971	G	G	G	G	G	G	Yes
53870	NZ_CP014971	C	C	C	C	C	C	Yes
53942	NZ_CP014971	A	A	A	A	A	A	Yes
53958	NZ_CP014971	T	T	T	G	T	T	No
54017	NZ_CP014971	G	G	G	G	G	G	Yes
54273	NZ_CP014971	T	T	T	T	T	T	Yes
54542	NZ_CP014971	G	G	G	G	G	G	Yes
54638	NZ_CP014971	A	A	A	A	A	A	Yes
54749	NZ_CP014971	C	C	C	C	C	C	Yes
54803	NZ_CP014971	T	T	T	T	T	T	Yes
54995	NZ_CP014971	C	C	C	C	C	C	Yes
55024	NZ_CP014971	T	T	T	T	T	T	Yes

Column width: Automatic

SNP information: By sample

Figure 9.9: *Counts of differences at a given position in the branches of the selected internal node. Top: SNP information As summary. Bottom: SNP information By sample.*

lower threshold value between 0 and the maximum value in the matrix. This results in a distinct coloring of the cells in the matrix which have a value less than the threshold.



		1	2	3	4	5	6	7	8	9	10
ERR274480	1	0	544	545	21	222	536	425	1416	27506	27496
ERR277216	2	544	0	9	543	554	66	537	1528	27617	27605
ERR277217	3	545	9	0	544	555	67	538	1529	27617	27605
ERR277220	4	21	543	544	0	221	535	424	1415	27505	27495
ERR277222	5	222	554	555	221	0	546	435	1426	27513	27503
ERR277224	6	536	66	67	535	546	0	529	1520	27607	27595
ERR277231	7	425	537	538	424	435	529	0	1029	27446	27434
ERR277235	8	1416	1528	1529	1415	1426	1520	1029	0	28307	28295
ERR277237	9	27506	27617	27617	27505	27513	27607	27446	28307	0	47
ERR277239	10	27496	27605	27605	27495	27503	27595	27434	28295	47	0

SNP Matrix Editor Settings

Contents

Comparison gradient

min max

Lower threshold: 0.00

Text format

Text size 12

Font SansSerif

□ Bold

Figure 9.10: *A SNP matrix.*

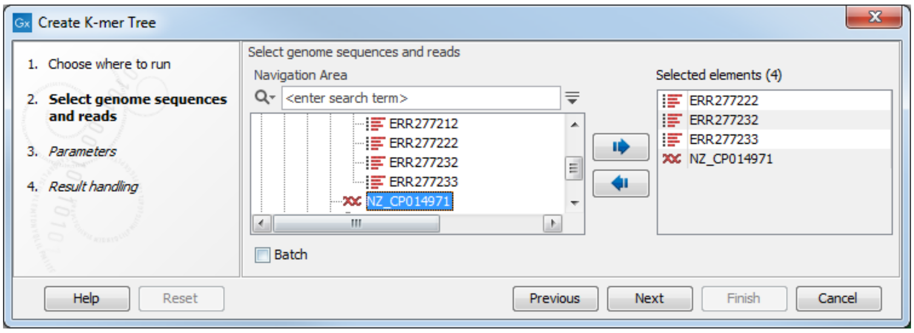
9.2 Create K-mer Tree

The Create K-mer Tree tool may be helpful for identification of the closest common reference across samples. The tool uses reads, single sequences or sequence list as input and creates a distance-based phylogenetic tree. If a sequence list has a read-group it will be treated as a set of reads, otherwise the tool will group the sequences in a sequence list based on their "Assembly ID" annotation or treat the sequences individually when no "Assembly ID" annotation has been assigned. To find out how to assign Assembly ID annotation, please see section 21. There are two ways to initiate creation of a k-mer tree: either from the Result Metadata Table (see chapter 19.2.2), or from the Toolbox.

To run the Create K-mer Tree from the toolbox:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Typing and Epidemiology** ( ) | **Create K-mer Tree** ( )

Input files can be specified step-by-step like shown in figure 9.11 or by selecting data recursively by right-clicking on the folder name and selecting **Add folder contents (recursively)**. If using the recursive option, remember to double check that files relevant for the downstream analysis are selected.



Create K-mer Tree

1. Choose where to run
2. **Select genome sequences and reads**
3. *Parameters*
4. *Result handling*

Select genome sequences and reads

Navigation Area

<enter search term>

ERR277212
ERR277222
ERR277232
ERR277233
NZ_CP014971

Selected elements (4)

ERR277222
ERR277232
ERR277233
NZ_CP014971

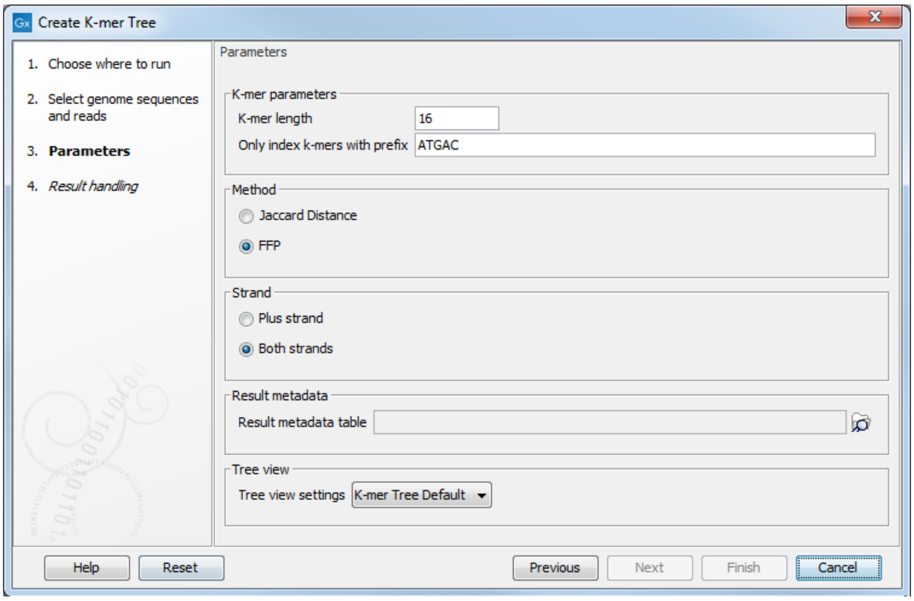
□ Batch

Figure 9.11: *Selection of individual reads and single sequences or sequence list to be included in the K-mer tree analysis.*

Specify the following parameters (figure 9.12):

- **K-mer parameters**
  - **K-mer length** is the fixed number (k) of DNA bases to search across.
  - **Only index k-mers with prefix** allows specification of the initial bases of the k-mer sequence to limit the search space. Reduction of prefix size increases the RAM requirements, and therefore decrease the search speed.

- **Method** may be specified by either of the two statistical methods: **Jaccard Distance** or **Feature Frequency Profile via Jensen-Shannon divergences (FFP)**. You can read more about the Jaccard Distance and FFP at https://en.wikipedia.org/wiki/Jaccard_index and https://en.wikipedia.org/wiki/Alignment-free_sequence_analysis, respectively.
- **Strand** may be specified as either only the Plus strand or Both strands.
- **Result metadata.** Specify location of the **Result metadata table** file.
- **Tree view.** Select a standard tree setting (i.e., None, K-mer Tree Default or SNP Tree Default) or your own custom tree setting. Read more on creating your customized tree settings: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.



K-mer parameters
K-mer length: 16
Only index k-mers with prefix: ATGAC

Method
○ Jaccard Distance
● FFP

Strand
○ Plus strand
● Both strands

Result metadata
Result metadata table:

Tree view
Tree view settings: K-mer Tree Default

Figure 9.12: *Various parameters may be set before generation of a K-mer tree.*

The K-mer trees are constructed using a Neighbour Joining method, which makes use of a distance function, either Jaccard Distance or Feature Frequency Profile via Jensen-Shannon divergences (FFP). In both cases, the distance can assume values between 0 (exactly same k-mer distribution) and 1 (completely different k-mer distribution).

Branch lengths depend on the distance function used. Specifically, if one sums up all the branch length of all the branches connecting two leaves, one can get the distance between the two organisms the leaves represent.

9.2.1 Visualization of K-mer Tree for identification of common reference

The k-mer tree below (figure 9.13) includes 46 samples and 44 *Salmonella* genomes. To identify a candidate common reference genome, the tree was visualized using the radial tree topology setting. The common reference is usually chosen as the genome sharing the closest common ancestor with the clade of isolates under study in the k-mer tree. In this case, a reference (acc no NC_011083) located in the centre region of the tree was selected as a common reference candidate.

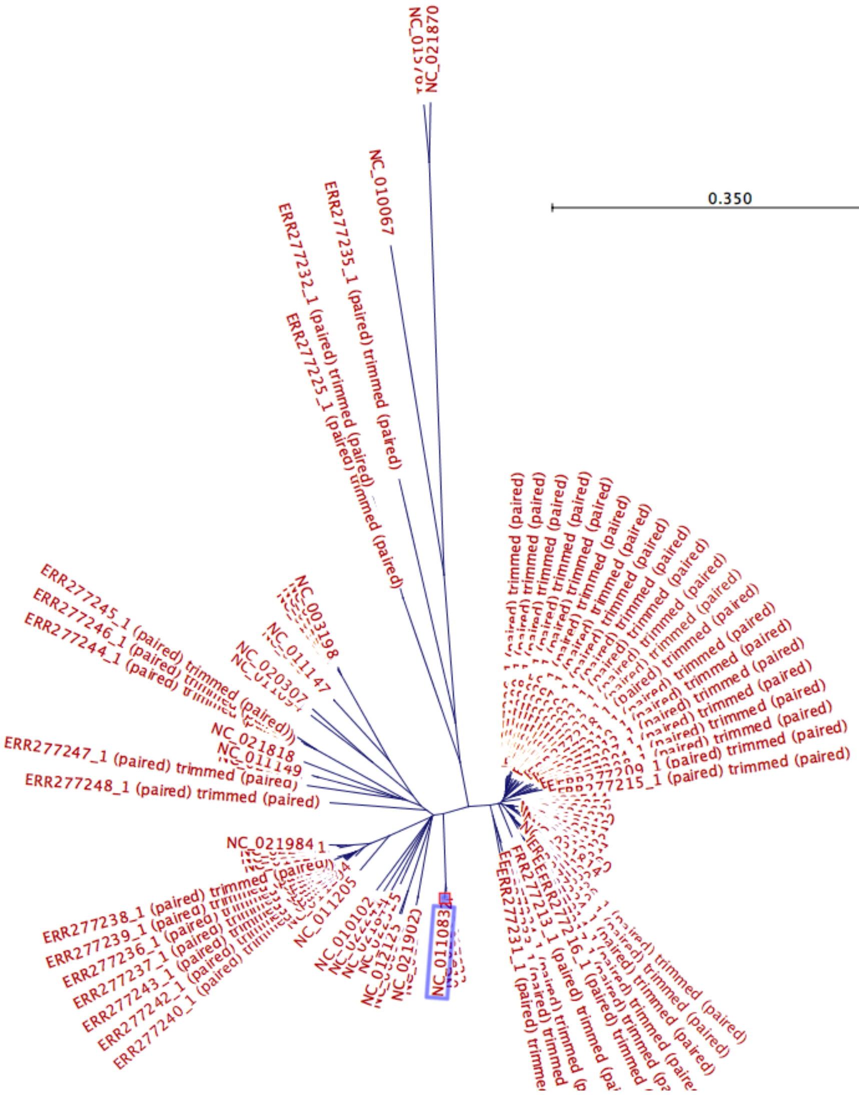

0.350
NC_021870
NC_010067
ERR277235_1 (paired) trimmed (paired)
ERR277232_1 (paired) trimmed (paired)
ERR277225_1 (paired) trimmed (paired)
NC_003198
NC_011147
NC_020307
ERR277245_1 (paired) trimmed (paired)
ERR277246_1 (paired)
ERR277244_1 (paired)
NC_021818
NC_011149
ERR277247_1 (paired) trimmed (paired)
ERR277248_1 (paired) trimmed (paired)
NC_021984
ERR277238_1 (paired) trimmed
ERR277239_1 (paired)
ERR277236_1 (paired)
ERR277237_1 (paired)
ERR277243_1 (paired)
ERR277242_1 (paired)
ERR277240_1 (paired)
NC_011205
NC_010102
NC_021902
NC_011083
ERR277215_1 (paired) trimmed (paired)
ERR277213_1 (paired) trimmed
ERR277216_1 (paired) trimmed
ERR277231_1 (paired) trimmed

Figure 9.13: *The created K-mer tree is visualized using the radial tree topology setting. The genome reference acc no NC_011083 situated in the center of the tree is selected as a common reference candidate.*

If the sequence lists (samples and reference genomes) used as input for a k-mer tree contains metadata, the information will be used to decorate the tree.

The scale bar refers to the branch lengths within the tree.

Note that the information in the Taxonomy column of the sequence list needs to be following this format: "Kingdom; Phylum; Class; Order; Family; Genus; Species".

The metadata will also be made available in the K-mer tree table view, where you can manually edit entries in the metadata fields by right clicking on it in the tabular view of the Sequence List. If samples and reference genomes share metadata columns with the same header, these columns will be merged in both the K-mer tree table view and tree view.

Learn more about the overall Tree Settings, including how to decorate trees with metadata, here https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.

Chapter 10

MLST Scheme Tools

The typing and characterization of pathogenic isolates plays an important part in epidemiology and outbreak analysis. MLST (Multilocus Sequence Typing) makes it possible to efficiently type strains against schemes with known isolates.

Whereas classic MLST analysis types isolates against a small set of gene fragments (typical 500bp fragments for a set of seven house-keeping genes), cgMLST (core-genome MLST) and wgMLST (whole-genome MLST) extends the analysis to thousands of loci, usually containing the complete coding gene sequences for the alleles for a given locus.

This section of the manual describes the MLST Scheme tools, which can be used to work with both cgMLST and wgMLST, as well as classic 7-gene schemes. The MLST Scheme typing can be applied either directly to the NGS reads of an isolate, or to an assembly of an isolate.

10.1 Getting started with the MLST Scheme tools

There are several ways to create MLST Schemes in the Microbial Genomics Module.

- **Create MLST Scheme**. Creates an MLST Scheme from sequence lists of reference genomes or assemblies with CDS annotations.
- **Download MLST Scheme**. Downloads existing MLST schemes from PubMLST.
- **Import MLST Scheme**. Imports an MLST scheme from a set of fasta files, sequence type info and optional locus metadata.

After a scheme has been obtained the following tools can be used together with the schemes.

- **Type With MLST Scheme**.
- **Add Typing Results to MLST Scheme**.

These tools are described in more detail in the following sections.

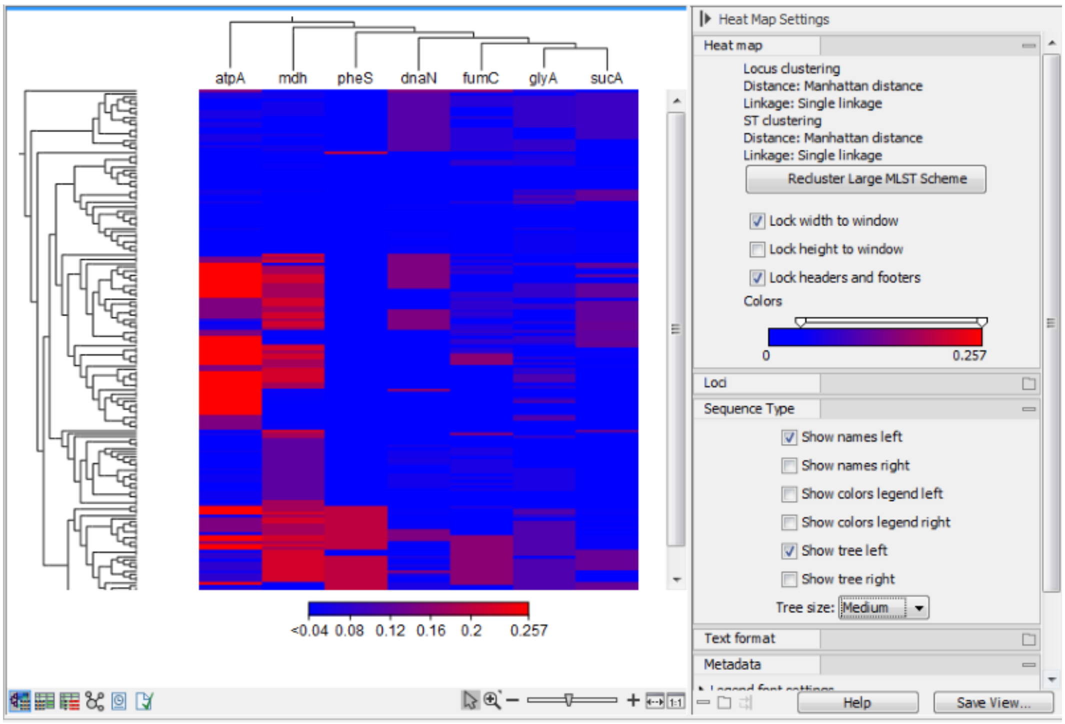
Finally, in order to be able to use the MLST Schemes outside of the Workbench, a MLST Scheme can be exported by clicking on the **Export** button and selecting **MLST Scheme**. To keep the data manageable, this will export the MLST Scheme into a single zip file containing two text files in tsv format, one with the sequence type definitions and sequence type metadata and the second file containing locus metadata, and one fasta file per locus, corresponding to the format used for importing MLST Schemes.

10.2 MLST Scheme Visualization and Management

A MLST Scheme contains information about:

- The loci that define the regions of interest.
- For each locus, a list of known alleles.
- A list of sequence types, where each sequence type is described by the alleles present at each locus (the profile of the sequence type).

The MLST Scheme has several views. Switching between the views of the scheme is done by clicking the buttons at the lower-left corner of the view.



Locus clustering
Distance: Manhattan distance
Linkage: Single linkage
ST clustering
Distance: Manhattan distance
Linkage: Single linkage

Recluster Large MLST Scheme

☑ Lock width to window
☐ Lock height to window
☑ Lock headers and footers

Colors 0 0.257

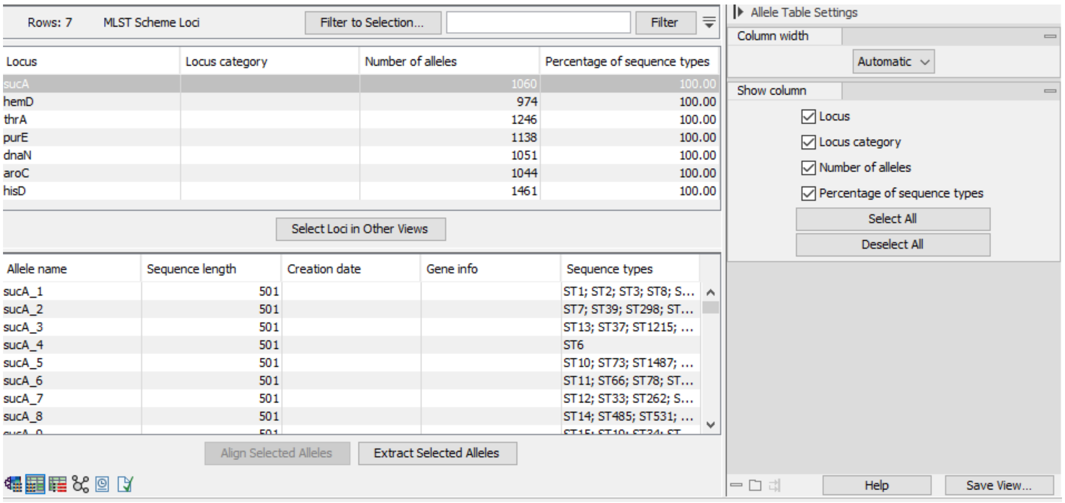
☑ Show names left
☐ Show names right
☐ Show colors legend left
☐ Show colors legend right
☑ Show tree left
☐ Show tree right

Tree size: Medium

Figure 10.1: *The MLST Scheme Heat Map view.*

The heat map view shows an overview of the scheme (figure 10.1), with the sequence types on the vertical axis, and the loci on the horizontal axis. Each cell in the heat map is colored according to the frequency of the allele in the given locus, that is, a value of 0.9 means 90% of the sequence types have this particular allele. Missing alleles will have a value of zero, alleles not present in any sequence type are not represented by the heat map view. The heat map can optionally be clustered based on the allele frequency. The clustering settings can be specified at scheme creation time, but it is also possible to use the **Recluster MLST scheme** button to update the clustering.

By right-clicking on the heat map, it is possible to either select sequence types or loci in other views or copy sequence or loci names to the clipboard.



Locus	Locus category	Number of alleles	Percentage of sequence types
sucA		1060	100.00
hemD		974	100.00
thrA		1246	100.00
purE		1138	100.00
dnaN		1051	100.00
aroC		1044	100.00
hisD		1461	100.00

Allele name	Sequence length	Creation date	Gene info	Sequence types
sucA_1	501			ST1; ST2; ST3; ST8; S...
sucA_2	501			ST7; ST39; ST298; ST...
sucA_3	501			ST13; ST37; ST1215; ...
sucA_4	501			ST6
sucA_5	501			ST10; ST73; ST1487; ...
sucA_6	501			ST11; ST66; ST78; ST...
sucA_7	501			ST12; ST33; ST262; S...
sucA_8	501			ST14; ST485; ST531; ...

Column width: Automatic

Show column
☑ Locus
☑ Locus category
☑ Number of alleles
☑ Percentage of sequence types

Figure 10.2: *The Allele table view.*

The allele table view (figure 10.2) has an upper table that lists the loci in the scheme. The table contains the following columns:

- **Locus**: the name of the locus
- **Locus category**: shows any virulence or resistance-gene related annotations.
- **Number of alleles**: the total number of alleles for this locus. Not all alleles may be part of a sequence type.
- **Percentage of sequence types**: shows how many of the sequence types have an allele in the given locus. For a strict core genome scheme, all of the sequence types contain all loci.

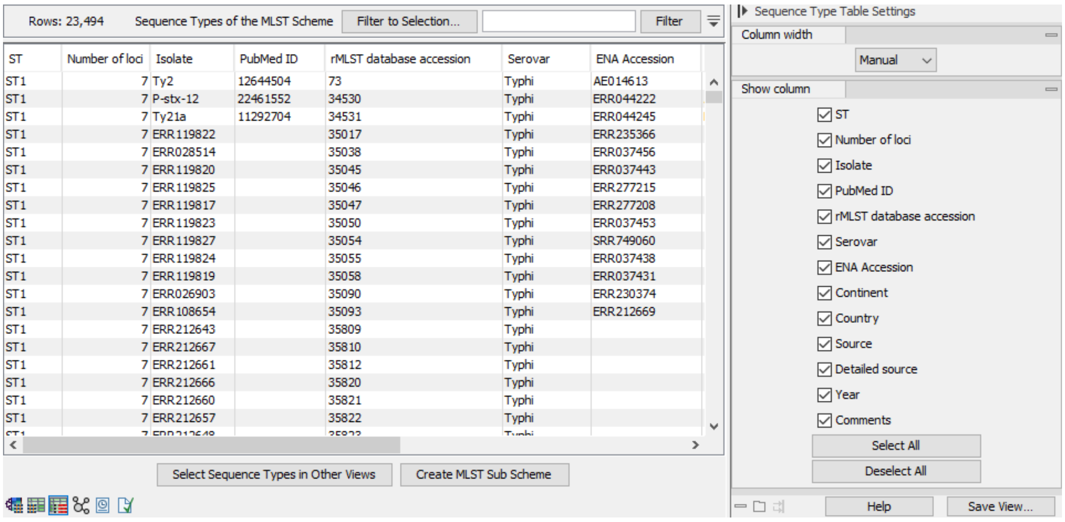
The lower table lists the alleles for the selected loci. It has the following columns:

- **Allele name**: the name of the allele.
- **Sequence length**: length in nucleotides.
- **Creation date**: when the allele was added.
- **Gene info**: AMR or virulence related information.
- **Sequence types**: the sequence types that contain this allele.

It is possible to **Align Selected Alleles**, which creates a new multiple sequence alignment view or to **Extract Selected Alleles**, which creates a sequence list with the alleles.

The Sequence Type table view (figure 10.3) shows the sequence types in the scheme. It always contains the following columns:

- **ST**: the name of the sequence type



Rows: 23,494 Sequence Types of the MLST Scheme

ST	Number of loci	Isolate	PubMed ID	rMLST database accession	Serovar	ENA Accession
ST1	7	Ty2	12644504	73	Typhi	AE014613
ST1	7	P-stx-12	22461552	34530	Typhi	ERR044222
ST1	7	Ty21a	11292704	34531	Typhi	ERR044245
ST1	7	ERR119822		35017	Typhi	ERR235366
ST1	7	ERR028514		35038	Typhi	ERR037456
ST1	7	ERR119820		35045	Typhi	ERR037443
ST1	7	ERR119825		35046	Typhi	ERR277215
ST1	7	ERR119817		35047	Typhi	ERR277208
ST1	7	ERR119823		35050	Typhi	ERR037453
ST1	7	ERR119827		35054	Typhi	SRR749060
ST1	7	ERR119824		35055	Typhi	ERR037438
ST1	7	ERR119819		35058	Typhi	ERR037431
ST1	7	ERR026903		35090	Typhi	ERR230374
ST1	7	ERR108654		35093	Typhi	ERR212669
ST1	7	ERR212643		35809	Typhi	
ST1	7	ERR212667		35810	Typhi	
ST1	7	ERR212661		35812	Typhi	
ST1	7	ERR212666		35820	Typhi	
ST1	7	ERR212660		35821	Typhi	
ST1	7	ERR212657		35822	Typhi	

Sequence Type Table Settings

Column width: Manual

Show column

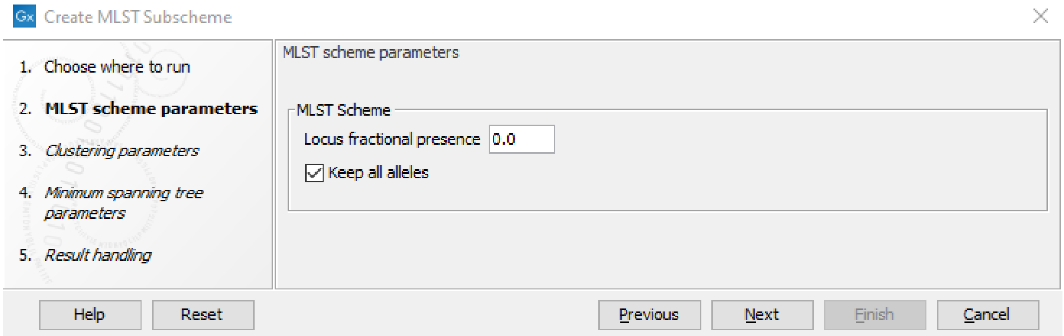
☑ ST
☑ Number of loci
☑ Isolate
☑ PubMed ID
☑ rMLST database accession
☑ Serovar
☑ ENA Accession
☑ Continent
☑ Country
☑ Source
☑ Detailed source
☑ Year
☑ Comments

Figure 10.3: *The Sequence Type table.*

- **Number of loci**: the number of loci, that are defined for this sequence type. Strict core genome schemes and classic 7-gene schemes will have the same number of loci for all sequence types.

Several other columns with arbitrary metadata information may be present as well.

At the bottom of the view, two buttons make it possible to **Select Sequence Types in Other Views** and to **Create Large Sub Scheme**.



Create MLST Subscheme

1. Choose where to run
2. **MLST scheme parameters**
3. *Clustering parameters*
4. *Minimum spanning tree parameters*
5. *Result handling*

MLST scheme parameters

MLST Scheme

Locus fractional presence 0.0

☑ Keep all alleles

Figure 10.4: *The Create MLST Subscheme options.*

The **Create Large Sub Scheme** has the same options (figure 10.4) as the other scheme creation tools, except for some additional options for pruning the scheme:

- **Locus fractional presence**: the fraction of sequence types required to have an allele specified for a given locus before the locus is added to the new scheme. For instance, a value of 0.95 would mean that the resulting scheme only contains loci present in at least 95% of the selected sequence types (a loose core genome scheme).
- **Keep all alleles**: if this option is deselected, only alleles that are part of at least one sequence type are retained. Alleles from discarded loci will always be removed.

Finally, the MLST Scheme also has a Minimum Spanning Tree view, which is the topic of the next section.

10.3 Minimum Spanning Trees

A minimum spanning tree is a tree connecting all nodes in a graph, in a way such that the sum of edge lengths is minimized, see figure 10.5.

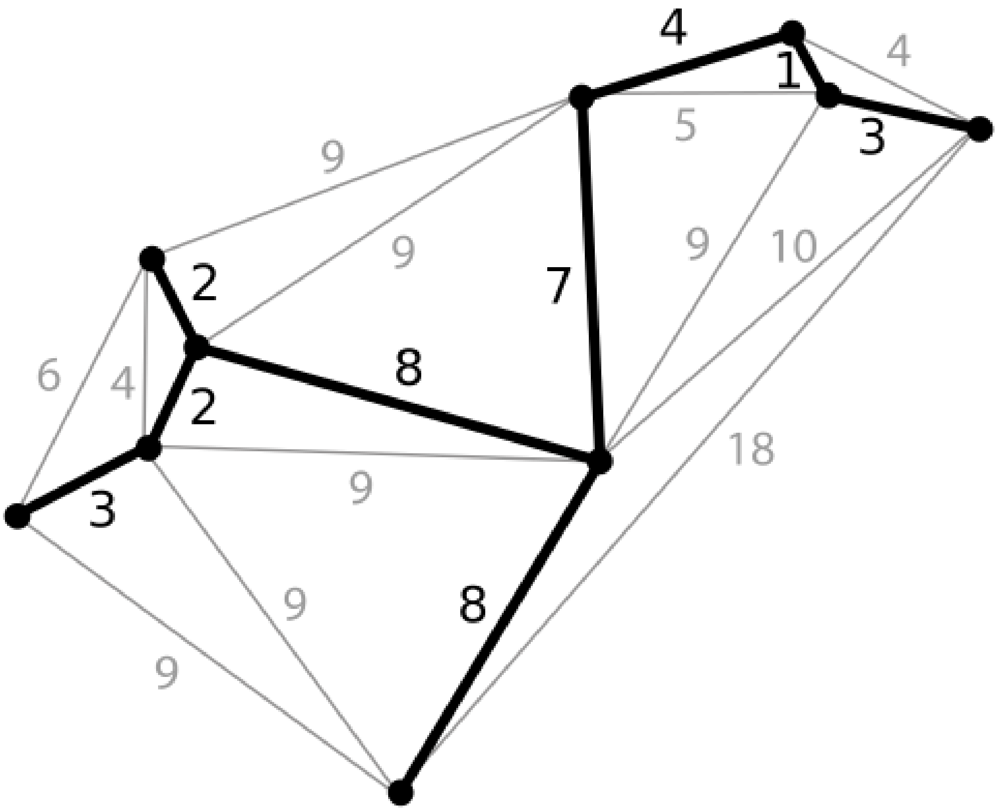


Figure 10.5: *Minimum Spanning Tree construction. (Public domain image from Wikimedia: Minimum spanning tree.svg.)*

Minimum spanning trees can be a bit counter-intuitive. Consider figure 10.6: The distance between A and B is not necessarily larger than the distance from A to C. But we do know that the distance between two nodes is greater or equal to the largest edge connecting them. (e.g. the distance between A and B is at least two, and the distance between A and C is at least two)

Minimum spanning trees are often used to visualize relationships between strains or isolates. But note that MST's are not unique - there are often many possible trees, especially for the classic 7-gene schemes, where there are only a very limited number of possible edge lengths. In order to break the ties when constructing the tree, our MST implementation favor creating connections to nodes that have many low-distance relations in the allelic distance matrix.

Minimum Spanning Trees can be created using the **Create MLST Scheme**, the **Download MLST Scheme**, the **Import MLST Scheme** tools or the **Create MLST Sub Scheme** button of an existing scheme.

10.3.1 The Minimum Spanning Tree view

It is possible to view the minimum spanning tree by selecting the MST icon ( ) at the bottom of the view, see figure 10.7.

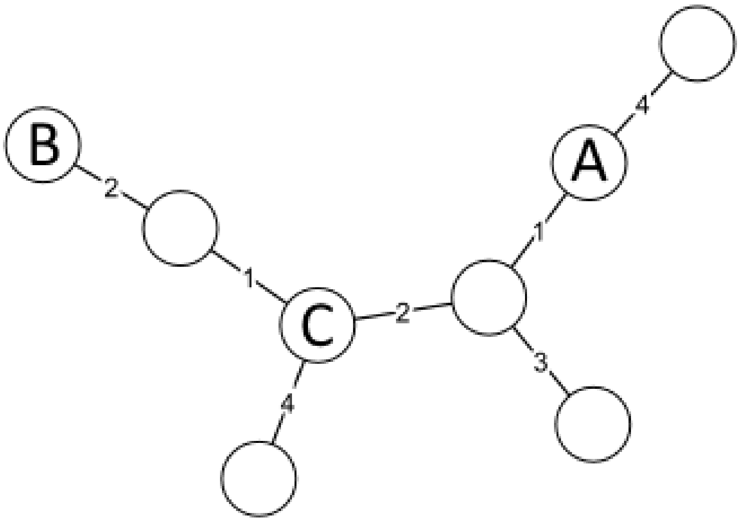


Figure 10.6: *Minimum Spanning Tree distances.*

An initial layout is calculated for a minimum spanning tree during the scheme creation, but it is possible to make changes to the layout.

If a change is made, the layout will be updated using a force-directed layout scheme: fictional forces are assigned to the tree nodes so that non-connected nodes will repel each other, while connected nodes will be held together with a spring force.

It is important to note that branch lengths in a force-directed layout may not be proportional to their ideal distance due to the repulsive force - in fact, the distance can be very different near heavy clusters.

10.3.2 Navigating the Tree view

It is possible to zoom in and out by pressing CTRL (or ⌘ on Mac) and using the scroll-wheel on the mouse.

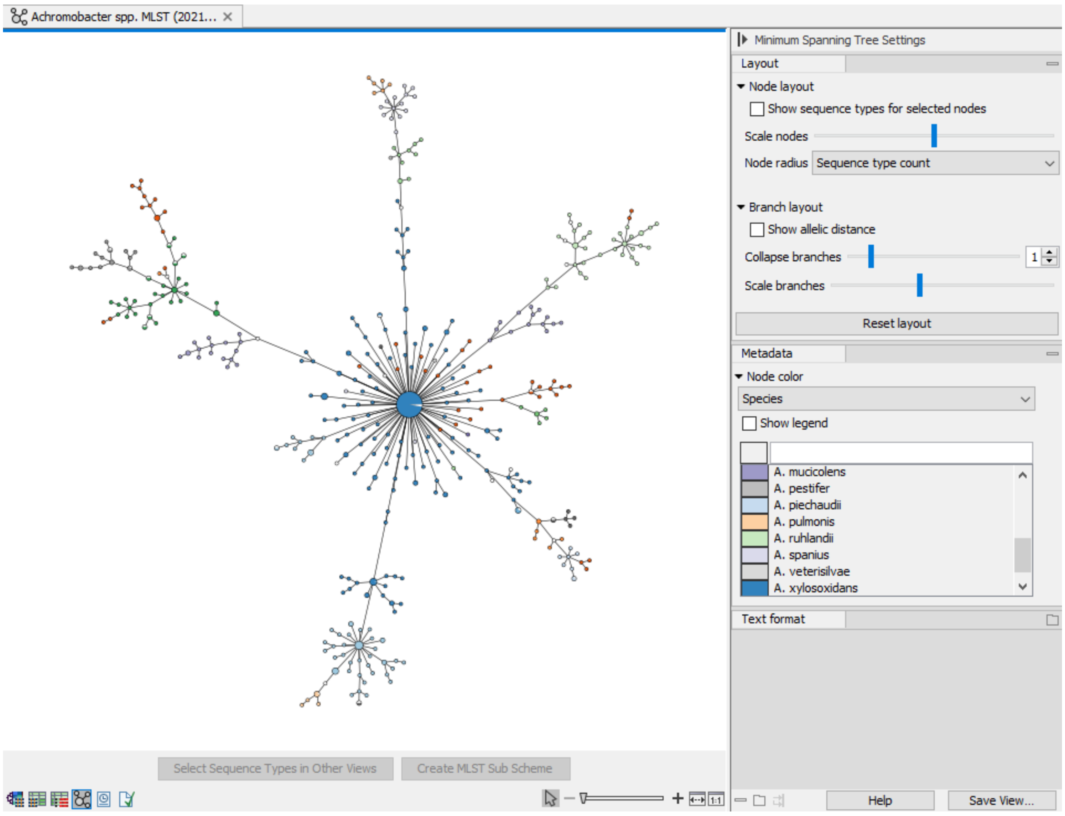
Nodes can be selected by clicking on them (which toggles them on and off), or by dragging the mouse to create a lasso selection (figure 10.8).

It is possible to clear the current selection by pressing on an empty region of the canvas.

When nodes are selected, they will stay in a fixed position. This can be helpful when manually adjusting the layout, for instance, to prepare the tree for publication (figure 10.9).

The following actions are available from the buttons at the bottom of the view:

- **Select Sequence Types in Other Views**: selected sequence types will be selected in other views that support it. Note that if nodes are collapsed, all the sequence types in a collapsed node will be selected in the other views.
- **Create MLST Sub Scheme**: This makes it possible to create a new scheme based on the


Achromobacter spp. MLST (2021...
Minimum Spanning Tree Settings
Layout
Metadata
Text format
Reset layout
Select Sequence Types in Other Views
Create MLST Sub Scheme
Help
Save View...

Node layout

□ Show sequence types for selected nodes

Scale nodes

Node radius: Sequence type count

Branch layout

□ Show allelic distance

Collapse branches: 1

Scale branches

Node color: Species

□ Show legend

🟪	A. mucicolens
⬜	A. pestifer
🟦	A. piechaudii
🟧	A. pulmonis
🟩	A. ruhlandii
⬜	A. spanius
⬜	A. veterisilvae
🟦	A. xylosoxidans

Figure 10.7: *The Minimum Spanning Tree view.*

selected sequence types.

10.3.3 The Layout panel

The following options are available: **Node layout**

- **Show sequence type names**: show the names of the sequence types for the nodes in the tree. If nodes are collapsed, only the first few names will be shown.
- **Scale nodes**: makes the nodes large or smaller.
- **Node radius**: either 'Sequence type count' or 'Isolate count'. A sequence type may have multiple isolates and metadata entries. This setting determines whether the node radius is based on the number of sequence types or isolates.

Branch layout

- **Show allelic distance**: shows the distance between different nodes in the tree. The distance is calculated as the number of loci where the allele assignment differs. Note that loci may have missing assignments. In this case, the distance calculation depends on the choice made when building the scheme (see section 13.1).
- **Collapse branches**: It is possible to reduce the complexity of the tree by clustering together nodes that are within a specific allelic threshold of each other. When setting a threshold, clusters will be formed where all nodes in a cluster are within the specified threshold to

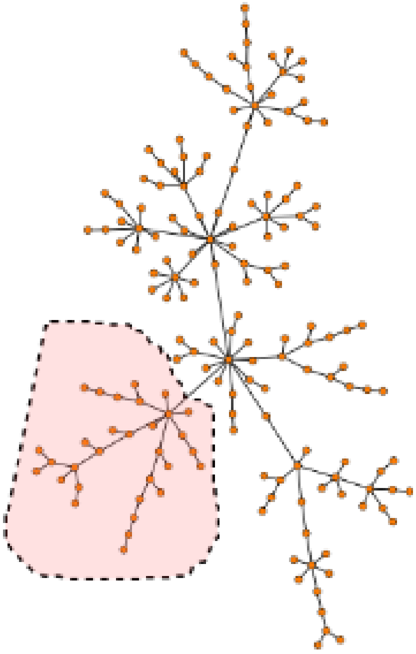


Figure 10.8: *Minimum Spanning Tree lasso selection.*

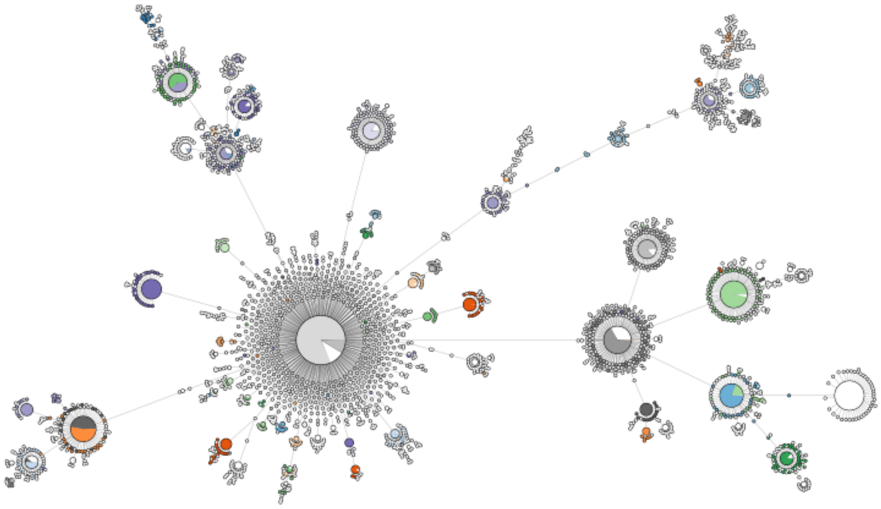


Figure 10.9: *Minimum Spanning Tree with manual layout modifications.*

at least one other node in the cluster (single-linkage clustering). See (figure 10.10) for an example of a tree where nodes have been collapsed.

- **Scale branches**: This parameter can be adjusted to make the branches longer. Note that the force-directed layout is primarily controlled by the repulsive force, so adjusting this parameter will not always have a proportional impact. Also note that due to the large span in allelic distances for cg- and wg-MLST schemes, the layout algorithm tries to fit an ideal branch length that is proportional to the square-root of the allelic distance.

- **Reset layout**: Pressing this button will reset the layout: this is done by first creating an initial radial layout, where no branches are crossed, and then applying a force-directed layout. If the graph is uncollapsed, pressing the **Reset layout** button will reset to the default layout created during scheme building.

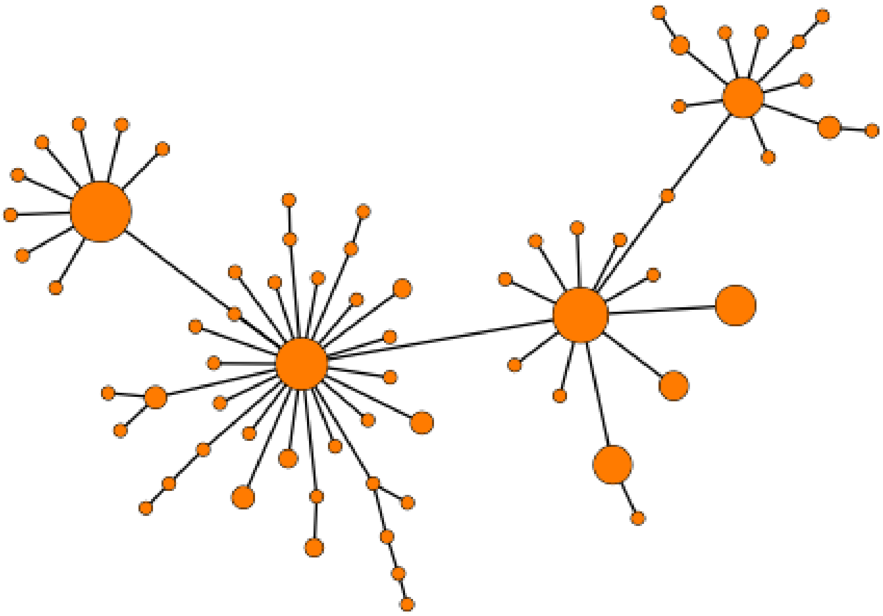


Figure 10.10: *Collapsing nodes in a Minimum Spanning Tree.*

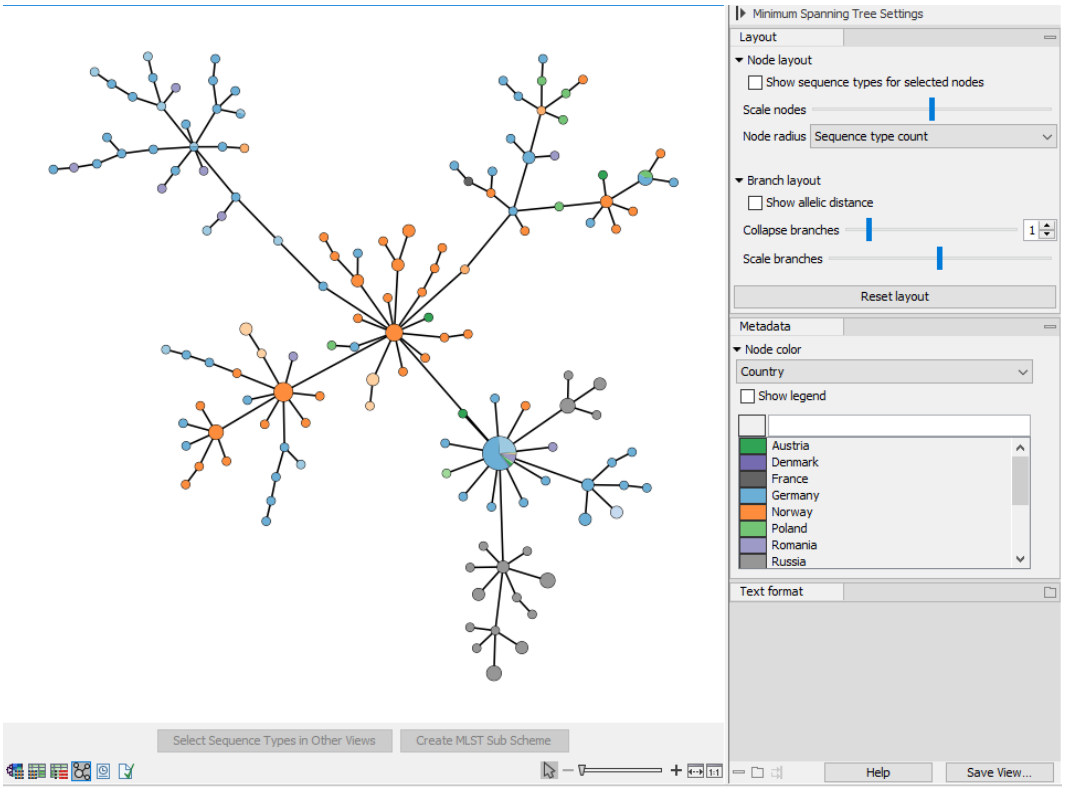
10.3.4 The Metadata panel

The metadata panel makes it possible to color node based on categorical metadata.

Collapsed nodes may have different metadata, in which case the fractional proportions of the different metadata categories will be shown as a pie-chart.

Note that uncollapsed nodes may have different multiple metadata values - this happens when a sequence type is associated with multiple metadata values, for instance from different isolates.

Note that when hovering over a node with the mouse, it is possible to see the distribution of metadata values at the status bar on the bottom of the application window.



□ Show sequence types for selected nodes

Scale nodes

Node radius: Sequence type count

Branch layout

□ Show allelic distance

Collapse branches: 1

Scale branches

Country

□ Show legend

🟩	Austria
🟪	Denmark
⬛	France
🟦	Germany
🟧	Norway
🟩	Poland
🟪	Romania
⬛	Russia

Figure 10.11: *Displaying metadata for Minimum Spanning Trees.*

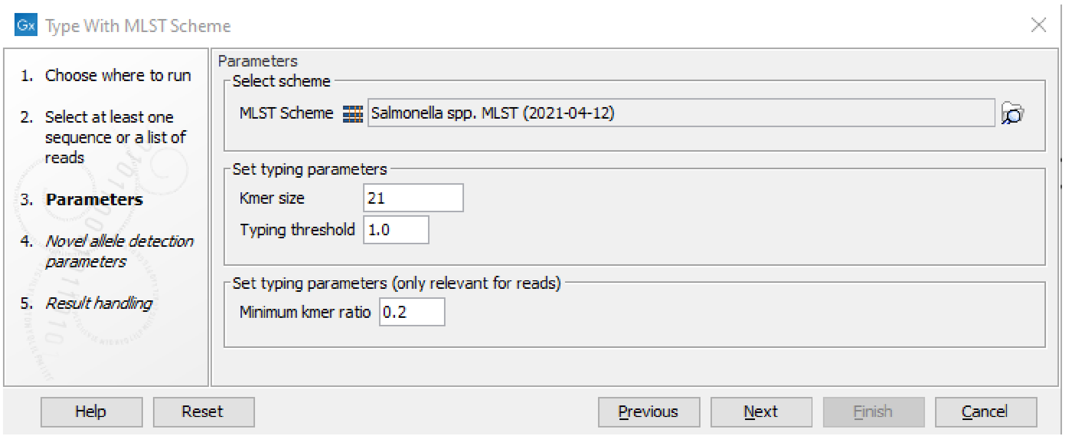
10.4 Type With MLST Scheme

The **Type With MLST Scheme** tool is used for assigning a sequence type to an isolate.

To run the **Type With MLST Scheme** tool choose:

Toolbox | **Microbial Genomics Module** ( ) | **Typing and Epidemiology** ( ) | **MLST Typing** ( ) | **Type With MLST Scheme** ( )

The tool takes a sequence list as input and will work with either raw NGS reads or an assembled genome. Note that if the input is raw NGS reads, and the tool reports multiple ambiguous sequence types, performing a standard De Novo Assembly might help to reduce noise and provide a more conclusive typing result.



1. Choose where to run
2. Select at least one sequence or a list of reads
3. **Parameters**
4. *Novel allele detection parameters*
5. *Result handling*

Parameters

Select scheme

MLST Scheme: Salmonella spp. MLST (2021-04-12)

Set typing parameters

Kmer size: 21

Typing threshold: 1.0

Set typing parameters (only relevant for reads)

Minimum kmer ratio: 0.2

Figure 10.12: *Specifying scheme and typing parameters.*

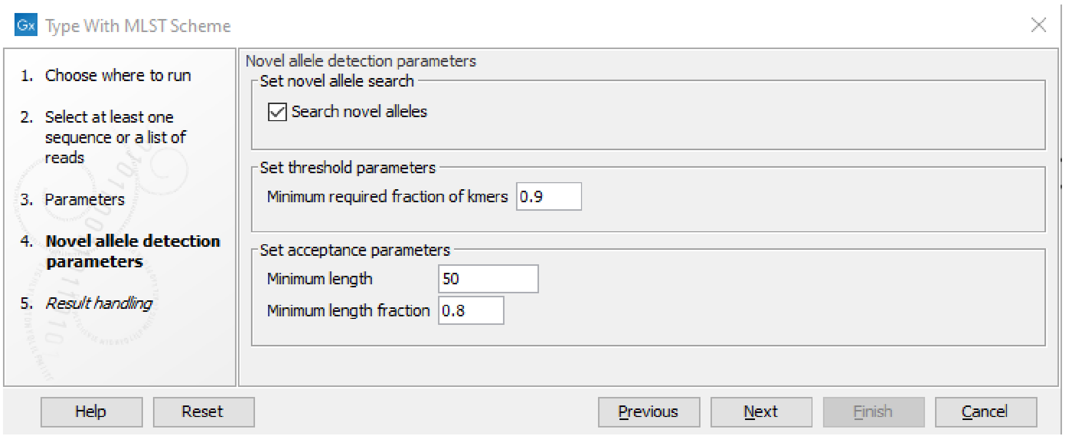
In the next dialog step (figure 10.12), specify the scheme and the typing parameters.

The tool works by comparing the kmers in the input to the kmers in the alleles for the different loci.

The **Kmer size** determines the number of nucleotides in the kmer - raising this setting might increase specificity at the cost of some sensitivity.

The **Typing threshold** determines how many of the kmers in a sequence type that needs to be identified before a typing is considered conclusive. The default setting of 1.0 means that all kmers in all alleles must be matched. Lowering the setting to 0.99 would mean that on average 99% of the kmers in all the alleles of a given sequence type must be detected before the sequence type is considered conclusive.

When working with reads, the Type With MLST Scheme tool works by classifying allele calls as high-confidence and low-confidence calls to remove alternative allele calls for the same locus. The **Minimum kmer ratio** threshold gives the possibility to tweak the balance between high-confidence and low-confidence allele calls, e.g. decreasing this number will result in more high-confidence allele calls and thus more ambiguity in how an ST is assigned to the sample, conversely increasing this number will result in fewer high-confidence calls and may lead to no allele being called for a particular locus, which can make sequence type assignments less confident. Specifically, the kmer ratio is calculated as the number of observations for the least occurring kmer in an allele divided by the average number of observations for all kmers.



Novel allele detection parameters

Set novel allele search

☑ Search novel alleles

Set threshold parameters

Minimum required fraction of kmers 0.9

Set acceptance parameters

Minimum length 50

Minimum length fraction 0.8

Figure 10.13: *Specifying novel allele detection parameters.*

The next step in the dialog determines how to handle novel alleles (figure 10.13): if the input isolate has loci with alleles that are not part of the scheme, it is possible to still detect the novel alleles. The novel alleles and the resulting new sequence type can then be added to the scheme using the **Add Typing Results to MLST Scheme** tool.

Novel alleles are detected as close hits to existing alleles in a locus. The **Minimum required fraction of kmers** determines how close a match must be: the default setting of 0.9 means that at least 90% of the kmers for an allele in a locus must be identified before the novel allele detection is initiated.

If the input to the tool is raw NGS reads, the tool will assemble the reads containing the kmers for the possible novel allele. If the input is already an assembled genome, the existing alleles for a locus will be mapped to the assembly to extract a novel allele.

After a candidate novel allele has been identified, it is aligned to the other alleles in the locus.

If the scheme has been built with the **Check codon positions** option of the **Create MLST Scheme** tool enabled (see section 13.1), or if the scheme was imported with a specified genetic code (see section 13.2), the start and stop codons in the novel allele sequence are then identified, and the sequence is then trimmed to the start and stop codons that most closely match the length of the existing alleles in the locus. Alleles that contain both a start and a stop codon at the beginning and end, respectively, and pass the acceptance parameters (see below) will be marked as Complete in the output table from the tool.

The acceptance parameters describe the final consistency check: the novel allele must not contain a stop codon, it must be at least the **Minimum length** in nucleotides and have at least a length of the specified **Minimum length fraction** of the shortest allele in the locus before it is accepted.

10.4.1 Type With MLST Scheme results

The **Type With MLST Scheme** tool outputs a report, summarizing the typing, and a MLST Typing Result element.

The report will contain overall information about the typing results and whether the typing was conclusive or not.

The typing information contains the following information:

- **Average kmer fraction**: for all alleles in a given sequence type, we calculate how many of the allele's kmers were detected. This number is the average fraction of the number of kmers detected in all these alleles.
- **Alleles identified**: how many alleles in the sequence type were identified, that is, all kmers in the allele were found in the sample.
- **Alleles called**: how many alleles in the sequence type had at least one kmer found in the sample.
- **Allele count**: the total number of alleles in the sequence type.

The sample information and scheme information contains various statistics about the input and scheme.

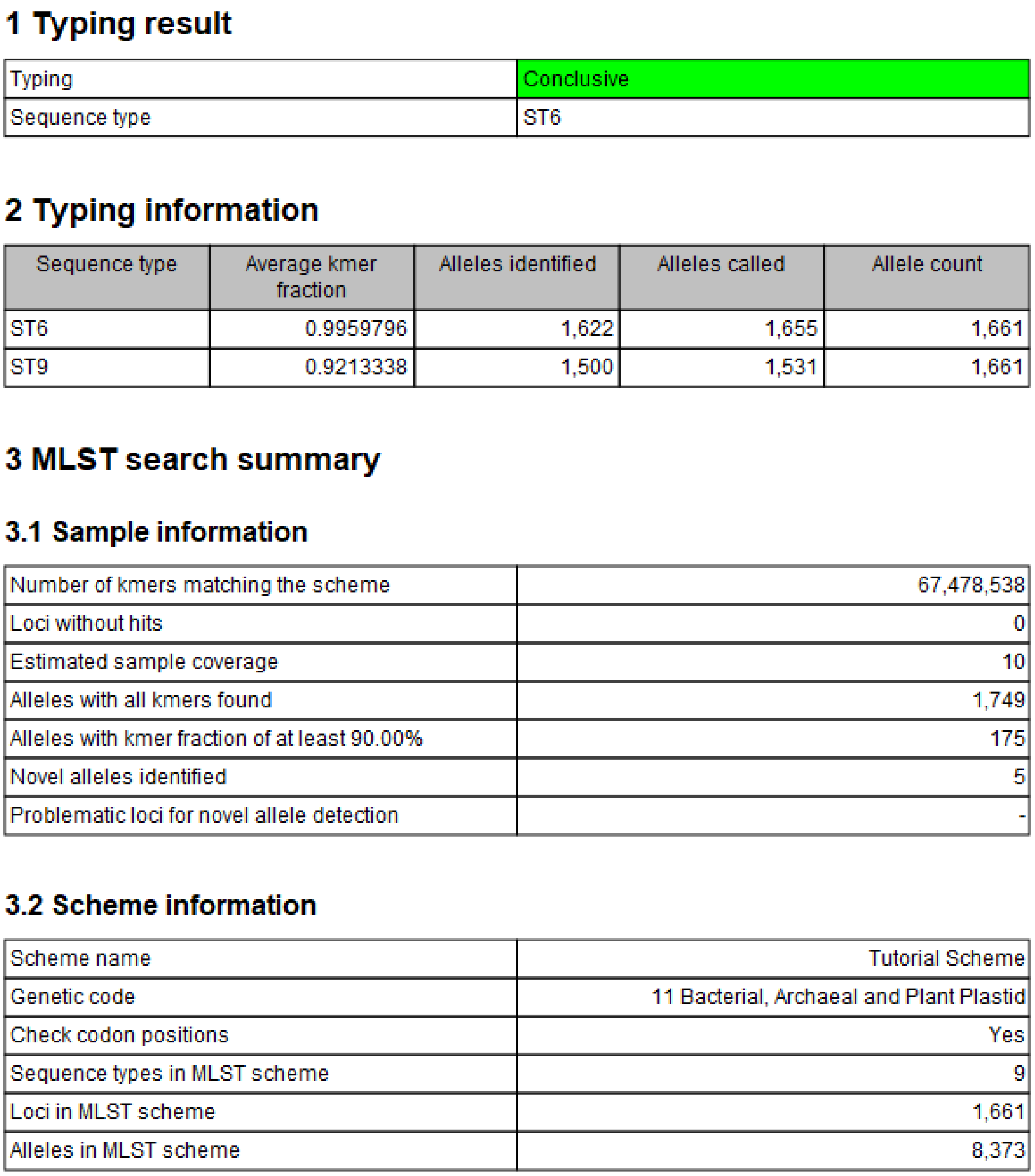
To add the report information to a Result Metadata Table, see section 19.2.3.

10.4.2 The MLST Typing Result element

The MLST Typing Result element contains several views. Switching between the views of the scheme is done by clicking the buttons at the lower-left corner of the view. The number of Sequence Types shown is limited to 100 or the number of Sequence Types in the scheme, depending on which is lower.

The sequence type table is a tabular view with information about how well the sample matched the sequence types in the scheme. It contains the following columns:

- **Sequence type**: name of the sequence type



1 Typing result

Typing	Conclusive
Sequence type	ST6

2 Typing information

Sequence type	Average kmer fraction	Alleles identified	Alleles called	Allele count
ST6	0.9959796	1,622	1,655	1,661
ST9	0.9213338	1,500	1,531	1,661

3 MLST search summary

3.1 Sample information

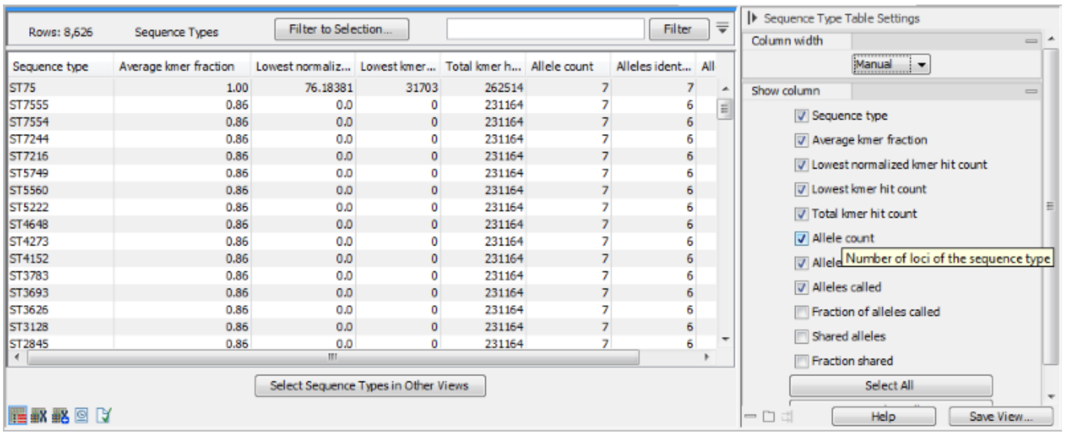
Number of kmers matching the scheme	67,478,538
Loci without hits	0
Estimated sample coverage	10
Alleles with all kmers found	1,749
Alleles with kmer fraction of at least 90.00%	175
Novel alleles identified	5
Problematic loci for novel allele detection	-

3.2 Scheme information

Scheme name	Tutorial Scheme
Genetic code	11 Bacterial, Archaeal and Plant Plastid
Check codon positions	Yes
Sequence types in MLST scheme	9
Loci in MLST scheme	1,661
Alleles in MLST scheme	8,373

Figure 10.14: *The Type With MLST Scheme report.*

- **Average kmer fraction**: for all alleles in a given sequence type, we calculate how many of the allele's kmers were detected. This number is the average fraction of the number of kmers detected in all these alleles.
- **Lowest normalized kmer hit count**: Normalized kmer hit count for the allele with the lowest normalized hit count of the sequence type.
- **Lowest kmer hit count**: Number of kmer hits for the allele with the fewest hits of the sequence type.
- **Total kmer hit count**: sum of kmer hits for all alleles of the sequence type
- **Allele count**: the total number of alleles in the sequence type.
- **Alleles identified**: the number of alleles in the sequence type where all kmers of the allele were found in the sample.



Rows: 8,626 Sequence Types

Sequence type	Average kmer fraction	Lowest normaliz...	Lowest kmer...	Total kmer h...	Allele count	Alleles ident...	All
ST75	1.00	76.18381	31703	262514	7	7	
ST7555	0.86	0.0	0	231164	7	6	
ST7554	0.86	0.0	0	231164	7	6	
ST7244	0.86	0.0	0	231164	7	6	
ST7216	0.86	0.0	0	231164	7	6	
ST5749	0.86	0.0	0	231164	7	6	
ST5560	0.86	0.0	0	231164	7	6	
ST5222	0.86	0.0	0	231164	7	6	
ST4648	0.86	0.0	0	231164	7	6	
ST4273	0.86	0.0	0	231164	7	6	
ST4152	0.86	0.0	0	231164	7	6	
ST3783	0.86	0.0	0	231164	7	6	
ST3693	0.86	0.0	0	231164	7	6	
ST3626	0.86	0.0	0	231164	7	6	
ST3128	0.86	0.0	0	231164	7	6	
ST2845	0.86	0.0	0	231164	7	6	

Sequence Type Table Settings

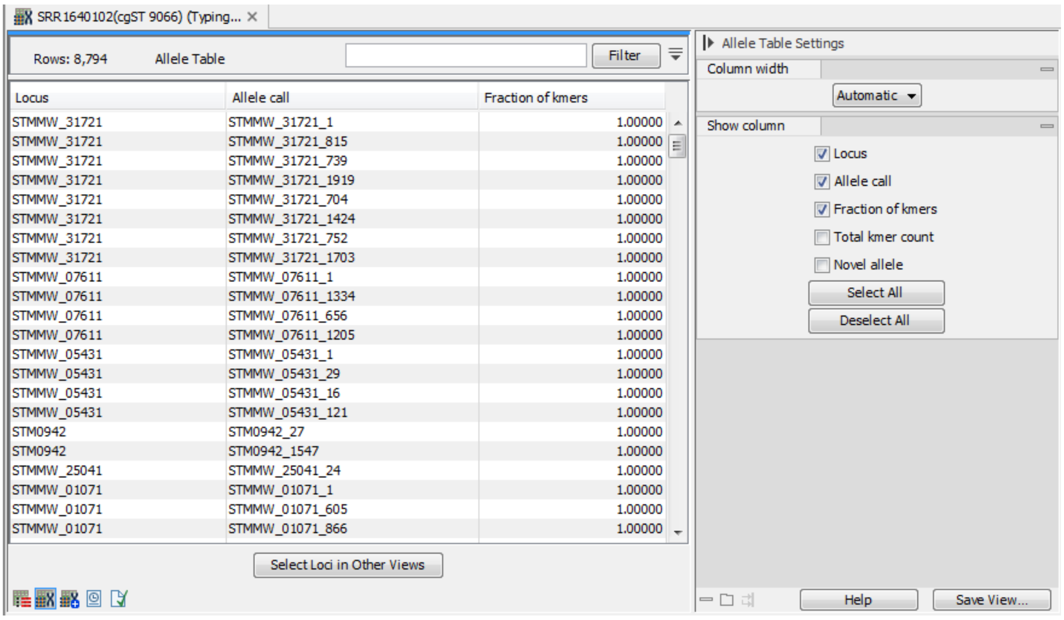
Column width: Manual

Show column

- ☑ Sequence type
- ☑ Average kmer fraction
- ☑ Lowest normalized kmer hit count
- ☑ Lowest kmer hit count
- ☑ Total kmer hit count
- ☑ Allele count
- ☑ Allele Number of loci of the sequence type
- ☑ Alleles called
- □ Fraction of alleles called
- □ Shared alleles
- □ Fraction shared

Figure 10.15: *The sequence type table for a MLST Typing Result.*

- **Alleles called**: the number of alleles in the sequence type with at least one kmer found in the sample.
- **Fraction of alleles called**: the ratio between alleles called and the allele count for the sequence type.
- **Shared alleles**: Number of alleles shared with the best scoring sequence type
- **Fraction shared**: Fraction of alleles shared with the best scoring sequence type



SRR1640102(cgST 9066) (Typing...

Rows: 8,794 Allele Table

Locus	Allele call	Fraction of kmers
STMMW_31721	STMMW_31721_1	1.00000
STMMW_31721	STMMW_31721_815	1.00000
STMMW_31721	STMMW_31721_739	1.00000
STMMW_31721	STMMW_31721_1919	1.00000
STMMW_31721	STMMW_31721_704	1.00000
STMMW_31721	STMMW_31721_1424	1.00000
STMMW_31721	STMMW_31721_752	1.00000
STMMW_31721	STMMW_31721_1703	1.00000
STMMW_07611	STMMW_07611_1	1.00000
STMMW_07611	STMMW_07611_1334	1.00000
STMMW_07611	STMMW_07611_656	1.00000
STMMW_07611	STMMW_07611_1205	1.00000
STMMW_05431	STMMW_05431_1	1.00000
STMMW_05431	STMMW_05431_29	1.00000
STMMW_05431	STMMW_05431_16	1.00000
STMMW_05431	STMMW_05431_121	1.00000
STM0942	STM0942_27	1.00000
STM0942	STM0942_1547	1.00000
STMMW_25041	STMMW_25041_24	1.00000
STMMW_01071	STMMW_01071_1	1.00000
STMMW_01071	STMMW_01071_605	1.00000
STMMW_01071	STMMW_01071_866	1.00000

Allele Table Settings

Column width: Automatic

Show column

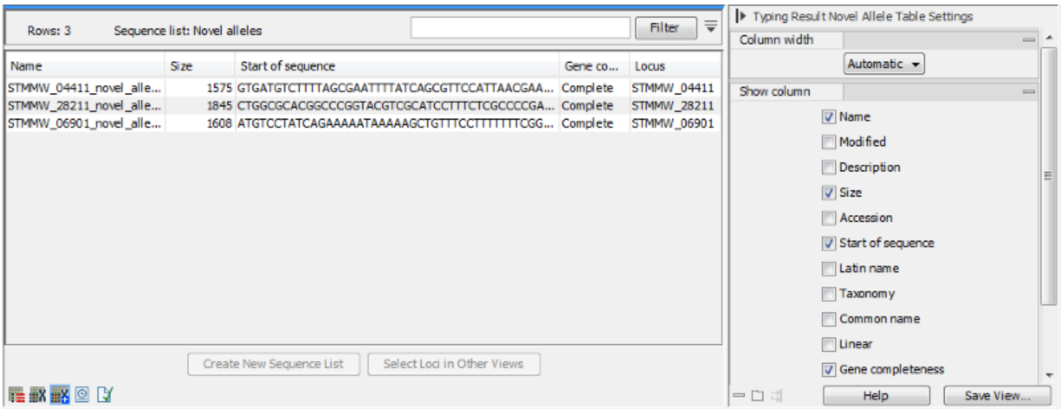
- ☑ Locus
- ☑ Allele call
- ☑ Fraction of kmers
- □ Total kmer count
- □ Novel allele

Figure 10.16: *The allele table for a MLST Typing Result.*

The allele table (figure 10.16) contains information about the alleles that were identified in the sample. It contains the following columns:

- **Locus**: the name of the locus.
- **Allele call**: the allele that was identified for that locus. Only the best allele is reported, but if multiple alleles are tied for the first place, they will all be reported.

- **Fraction of kmers**: the fraction of kmers of the allele that were found in the sample.
- **Total kmer count**: total number of kmer hits for the allele.
- **Novel allele**: contains the string 'Novel' for novel alleles, otherwise this field is left blank.



Rows: 3 Sequence list: Novel alleles

Name	Size	Start of sequence	Gene co...	Locus
STMMW_04411_novel_alle...	1575	GTGATGTCTTTTAGCGAATTTTATCAGCGTTCCATTAACGAA...	Complete	STMMW_04411
STMMW_28211_novel_alle...	1845	CTGGCGCACGGCCCGGTACGTCGCATCCTTTCTCGCCCCGA...	Complete	STMMW_28211
STMMW_06901_novel_alle...	1608	ATGTCCTATCAGAAAAATAAAAAGCTGTTTCCTTTTTTTCGG...	Complete	STMMW_06901

Typing Result Novel Allele Table Settings

Column width: Automatic

Show column

☑ Name
☐ Modified
☐ Description
☑ Size
☐ Accession
☑ Start of sequence
☐ Latin name
☐ Taxonomy
☐ Common name
☐ Linear
☑ Gene completeness

Figure 10.17: *The novel allele view for a MLST Typing Result.*

The novel allele view (figure 10.17) contains the novel alleles that were detected (if searching for novel alleles was enabled during the typing).

It is a sequence list, and it is possible to extract the complete sequences using the **Create New Sequence List** button.

The **Gene completeness** column is the only non-standard sequence list column: if a novel allele starts with a start codon and ends with a stop codon it is considered complete. Note that all novel alleles found with a scheme without a translation code will be incomplete.

10.5 Add Typing Results to MLST Scheme

After typing an isolate, it is possible to add the information to the MLST Scheme. There are several different possibilities when adding a typing result:

The typing result may have matched an existing sequence type completely. In this case, it is still possible and useful to add the typing result to the scheme, in order to add additional isolate metadata for the sequence type which may be from metadata annotations on the sequences or from a metadata table.

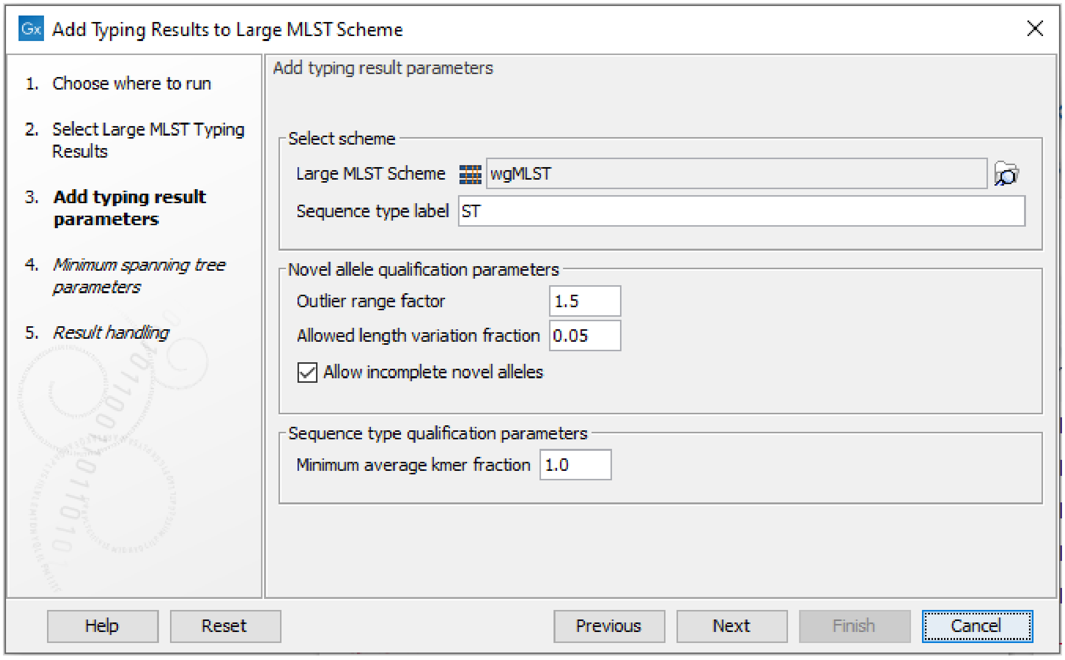
The typing result may have matched existing alleles in the scheme but in a new combination not present in any of the existing sequence types. In this case, the typing result will simply be added as a new sequence type.

The typing result may introduce new (novel) alleles to the scheme. In this case, both a new sequence type and one or more alleles are added to the scheme.

To run the **Add Typing Results to MLST Scheme** tool choose:

> **Toolbox** | **Microbial Genomics Module** () | **Typing and Epidemiology** () | **MLST Typing** () | **Add Typing Results to MLST Scheme** ()

After selecting the MLST Typing result to add, the next step is to set up the **Add typing result parameters** (figure 10.18):



Add Typing Results to Large MLST Scheme

1. Choose where to run
2. Select Large MLST Typing Results
3. **Add typing result parameters**
4. *Minimum spanning tree parameters*
5. *Result handling*

Add typing result parameters

Select scheme

Large MLST Scheme: wgMLST

Sequence type label: ST

Novel allele qualification parameters

Outlier range factor: 1.5

Allowed length variation fraction: 0.05

☑ Allow incomplete novel alleles

Sequence type qualification parameters

Minimum average kmer fraction: 1.0

Figure 10.18: *Add typing result parameters.*

- **MLST Scheme**: the scheme that the typing results will be added to. Adding the typing results will not modify the original scheme, but create a new copy with the added types.
- **Outlier range factor**: the allele length outlier definition in terms of the interquartile range of the length distribitution of alleles in a locus. Novel alleles outside this range will not be added to the scheme.
- **Allowed length variation fraction**: The allowed length variation of a novel allele with respect to the median length of alleles in a locus. This allows adding novel alleles with minor length variations irrespective of the outlier definition.
- **Allow incomplete novel alleles**: whether only complete novel alleles (containing both start and stop codon) should be allowed. If incomplete novel alleles are not allowed, a sequence type with incomplete alleles for a locus will be added with missing alleles for that locus. Classic 7-gene schemes typically contain partial gene fragments, and in this case incomplete novel alleles should be allowed.
- **Minimum average kmer fraction**: if this value is larger than zero, a typing result will only be added if the isolate was sufficiently similar to an already existing sequence type in the scheme - or to put it differently - at least one of the sequence types in the MLST Typing Result must have an Average kmer fraction larger than this threshold. Note, that when typing against an empty scheme, this value must be set to zero, to allow for the sequence type to be added. This option is mostly useful when adding a large number of isolates in bulk without manually inspecting them.

After adding new sequence types it is necessary to recreate the Minimum Spanning Tree. The options are the same as described in the **Download MLST** section (section 13.2)

10.6 Identify MLST Scheme from Genomes

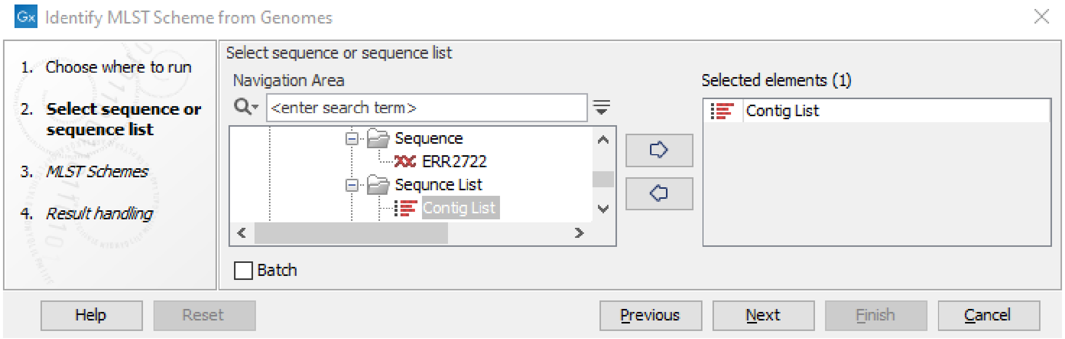
This section describes how to perform the identification of the relevant MLST Scheme for a genome sequence or list of genome sequences.

This tool can be used before running the Type With MLST Scheme tool in case you are working with a sample containing a single or multiple unknown species, as in the Type among Multiple Species workflow.

To run the **Identify MLST Scheme from Genomes** tool choose:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Typing and Epidemiology** ( ) | **MLST Typing** ( ) | **Identify MLST Scheme from Genomes** ( )

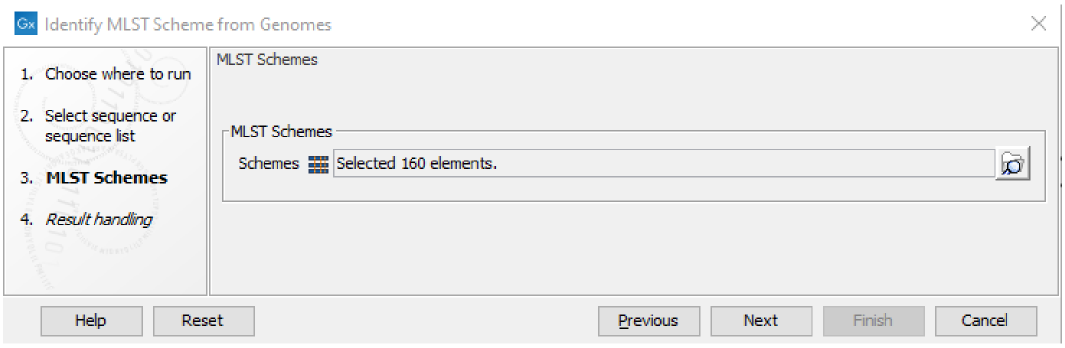
The input to the tool is a sequence, or a sequence list (figure 10.19).



Selected elements (1)
Contig List

Figure 10.19: *Select relevant genome sequence or sequence list.*

The next step is to select as many MLST schemes as necessary to identify the species present in the input sample (figure 10.20).



Schemes Selected 160 elements.

Figure 10.20: *Select relevant MLST scheme(s) to search among.*

To identify the best matching scheme, the tool identifies the 10 most prevalent loci, i.e. loci that occur in most or all of the sequence types. If fewer loci are available, the tool will base the identification on these, thus the tool also works for classic 7-gene MLST schemes, given that they are in the MLST Scheme format.

The k-mers for all alleles for these most prevalent loci are then determined, and the provided references are checked for their presence.

The output of this tool is the MLST scheme that best matches the sequences analyzed. To add the obtained best match to a Result Metadata Table, see section 19.2.3.

The tool will not produce an output if no scheme could be uniquely identified.

Part V

Functional and Drug Resistance Analyses

Chapter 11

Functional Analysis

Two of the most widely used definitions of biological function are available in the form of the Gene Ontology (GO) and Pfam databases. While GO is a hierarchy of higher-level functional categories, Pfam (Protein families) classifies proteins into families of related proteins with similar function.

Several tools are available for functional analysis. From a whole metagenome shotgun sequencing dataset as reads, the first step is to assemble the reads using the **De Novo Assemble Metagenome** tool (see section 3). The resulting contigs can then be annotated with coding sequences (CDS) using the **Find Prokaryotic Genes** tool. Given a set of contigs with CDS annotations, the **Annotate CDS with Best BLAST Hit**, the **Annotate CDS with DIAMOND Hits** and the **Annotate CDS with Pfam Domains** tools can be used to annotate all CDS in the annotated contigs with BLAST or DIAMOND hits or Pfam protein families and GO terms, respectively. The database needed for GO annotation can be downloaded using the **Download GO Database** tool, while the Pfam database can be downloaded using the built-in **Download Pfam Database** tool and BLAST databases can be downloaded or created using the built-in **Download BLAST Dabases** and **Create BLAST Database** tools.

Once the contigs are annotated with Pfam annotation, GO terms and/or BLAST hits, the next step will often be to map the original reads back to the annotated contigs, using the built-in **Map Reads to Reference** tool, in order to be able to assess the abundance of the functional annotations. This last step is performed using the **Build Functional Profile** tool (see section 11.7).

All tools described above should be run independently for individual samples (or batched), resulting in a functional profile for each sample. A set of functional profiles can then be joined using the **Merge Abundance Tables** tool (see section 6.1). The functional profile of multiple samples can now be visualized and compared as described in section 4.3.2.

11.1 Find Prokaryotic Genes

The **Find Prokaryotic Genes** tool allows you to annotate a DNA sequence with CDS information. The tool is currently for use with near-complete single prokaryotic genomic and metagenomic data.

The tool creates a gene prediction model from the input sequence, which estimates GC content, conserved sequences corresponding to ribosomal binding sites, start and stop codon usages, and a statistical model (namely, an Interpolated Markov Model) for estimating the probability of a sequence to be part of a gene compared to the background. The model is then used to predict coding sequences from the input sequence. Note that this tool is inspired by Glimmer 3 (see https://ccb.jhu.edu/papers/glimmer3.pdf).

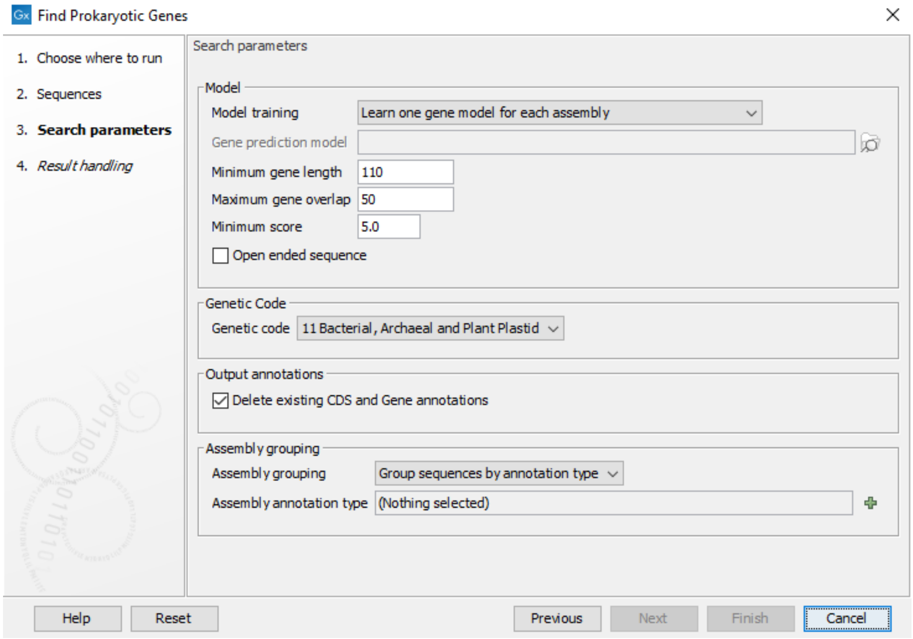
To maximize the gene prediction accuracy, the gene models should be trained on sequences that belong to the same species or to similar ones. When the input consists of sequences originating from multiple organisms, it is recommended to build a gene model for each organism by choosing the "Learn one gene model for each assembly" option. In assembly grouping, there are multiple options for specifying what should be considered an assembly. For example, when downloading assemblies from the Download Custom Microbial Reference Database tool and the prokaryotic databases from the Download Curated Microbial Reference Database tool, the "Assembly ID" column will be automatically populated and can be used for grouping, see section 21. When assembly information is not known, for example when the input consists of de novo assemblies, the option "Each input element is one assembly can be used". When working with de novo assembled metagenomics sequences, the Bin Pangenomes by Sequence and Bin Pangenomes by Taxonomy tools can be used to group sequences into bins whose sequences are likely to come from the same organism.

To start the analysis, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Find Prokaryotic Genes** ( )

In the first dialog, select input sequences. The input should consist of one or few contigs from the same species. If several sequences are provided as input, the model training can be used to specify if the tool should build a separate model for each assembly. The tool can also be run in batch mode.

In the second dialog (figure 11.1), it is possible to configure the tool.



Find Prokaryotic Genes

1. Choose where to run
2. Sequences
3. **Search parameters**
4. *Result handling*

Search parameters

Model

Model training: Learn one gene model for each assembly

Gene prediction model:

Minimum gene length: 110

Maximum gene overlap: 50

Minimum score: 5.0

□ Open ended sequence

Genetic Code

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Output annotations

☑ Delete existing CDS and Gene annotations

Assembly grouping

Assembly grouping: Group sequences by annotation type

Assembly annotation type: (Nothing selected)

Figure 11.1: *Configuring the Find Prokaryotic Genes tool*

Model

- Learn one gene model: Learns a single model from the data. Assumes that the sequences come from one organism or a group of closely related organisms.
- Learn one gene model for each assembly: Learns a model for each assembly or bin. This option should be used when assemblies can be clearly distinguished, for example when they are separated with one assembly per sequence list or are assigned with with an ID in the Assembly ID column as is the case with output from Bin Pangenomes by Taxonomy, Download Custom Microbial Reference Database and the prokaryotic databases from the Download Curated Microbial Reference Database tool.
- Use a previously trained model and use its default parameters: This option allows to choose a model that has been previously trained and run the analysis with the same parameters used when training the model.
- Use a previously trained model: This option allows to choose a model that has been previously trained. It also allows to modify some parameters.

In all but one of this option, the following parameters can be modified:

- Minimum gene length: in bp, excluding start and stop codons.
- Maximum gene overlap: in bp
- Minimum score: Putative genes with a score below this value will be ignored. The value of a gene score depends on how well the sequence of the gene matches the model. It is computed by taking into account how much the sequence is typical of a coding region (as opposed to background noise or the same coding region read in a different frame), of the prevalence of the start codon, and of the presence of a putative ribosomal binding site near the start codon.
- Open ended sequence: check this option to annotate open-ended sequences, which is particularly useful for annotating small contigs.

Genetic Code The genetic code to use (default to bacterial). This genetic code is used to determine which stop codons should be used and to compute a background distribution for amino-acid usage.

Output annotations Delete Existing CDS and Gene Annotations. This is selected by default in order to avoid having many duplicate annotations. Unchecking is useful if one wants to compare the results with other annotations.

Assembly Grouping

- Each sequence is one assembly: Each sequence in the input elements is treated as one assembly.
- Each input element is one assembly: Each input element is treated as one assembly regardless of annotation types and number of sequences in the input element.
- Group sequences by annotation type: This option allows to choose an annotation type. Each unique label in the input is then treated as one assembly.

The tool will output a copy of the input sequence with CDS and Gene annotations. It is possible to save the gene model(s) used for the analysis when the option "Learn one gene model" was selected earlier. This model can then be reused to annotate other input sequences by setting the "Model Training" option to "Use a previously trained model" or "Use a previously trained model and use its default parameters".

11.2 Annotate with BLAST

The **Annotate with BLAST** tool allows you to annotate a DNA sequence using a set of either protein reference sequences or nucleotide sequences. This tool can be used on sequences without any pre-existing annotations: it is not necessary to annotate the DNA sequences with genes or coding regions.

The tools can be used for various purposes, e.g. transferring annotations from a known reference, annotate the presence of AMR or virulence markers in a genome, or to filter contigs or sequences based on the presence of a set of genes.

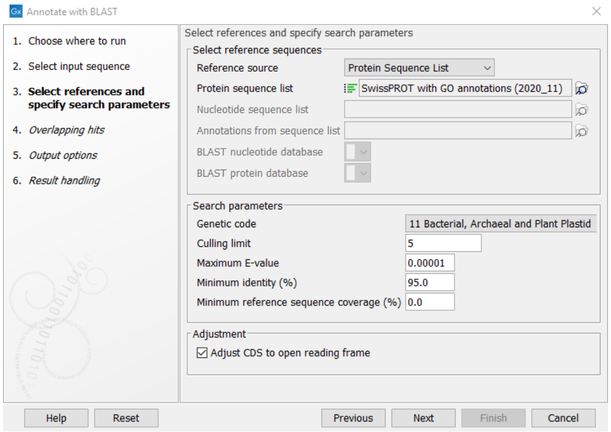
If the reference sequences are protein sequences, the **Annotate with DIAMOND** tool may be used instead and is a faster option.

If the input sequences are already annotated with CDS annotations, it is also possible to use the **Annotate CDS with Best BLAST Hit** and **Annotate CDS with Best DIAMOND Hit** tools - see section 11.4 for more information.

To start the analysis, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Annotate with BLAST** ( )

The first wizard step (figure 11.2), specifies the reference and search parameters.



Annotate with BLAST

1. Choose where to run
2. Select input sequence
3. **Select references and specify search parameters**
4. *Overlapping hits*
5. *Output options*
6. *Result handling*

Select references and specify search parameters

Select reference sequences

Reference source: Protein Sequence List

Protein sequence list: SwissPROT with GO annotations (2020_11)

Nucleotide sequence list:

Annotations from sequence list:

BLAST nucleotide database:

BLAST protein database:

Search parameters

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Culling limit: 5

Maximum E-value: 0.00001

Minimum identity (%): 95.0

Minimum reference sequence coverage (%): 0.0

Adjustment

☑ Adjust CDS to open reading frame

Figure 11.2: *Selecting references and specifying search parameters*

The following sources can be used to annotate the input sequences:

- **Protein sequence list**. The nucleotide input query will be searched against the sequences in the protein sequence list. The nucleotide input will be translated using the chosen genetic code. If the reference protein sequence list contains metadata, this metadata will be transferred to the resulting annotations on the input query sequence.

- **Nucleotide sequence list**. The nucleotide input query will be searched against the sequences in the nucleotide sequence list. If the reference nucleotide sequence list contains metadata, this metadata will be transferred to the resulting annotations on the input query sequence.

- **CDS Annotations (blastx)**. This option uses a nucleotide sequence source with existing annotations as a source. All annotations are extracted, and translated to a protein database, which is searched similar to the **Protein sequence list** option. All qualifiers on the detected source annotations are transferred to the input query sequence.

- **All Annotations (blastn)**. This option uses a nucleotide sequence source with existing annotations as a source. All annotations are extracted and searched similar to the **Nucleotide sequence list** option. All qualifiers on the detected source annotations are transferred to the input query sequence.
- **BLAST nucleotide database**. BLAST databases can be created using the **Create BLAST database** tool. This option works similar to the **Nuclotide sequence list** option, but can be faster, since the database can be reused. When using this option, the name and description of detected reference sequences are transferred to the input query sequence.
- **BLAST protein database**. BLAST databases can be created using the **Create BLAST database** tool, or downloaded using the **Download BLAST database** tool. This option works similar to the **Protein sequence list** option, but can be faster, since the database can be reused. When using this option, the name and description of detected reference sequences are transferred to the input query sequence.

As can be seen above, metadata (such as GO terms and taxonomy information) is handled differently depending on the database source:

- **Protein / nucleotide sequence list**. Sequence lists may contain metadata, which can be inspected in the table view of the sequence list. Such metadata is transferred to the annotations created by this tool.
- **CDS / all annotations**. Annotations are transferred together with any metadata qualifiers the annotations contain.
- **BLAST protein / nucleotide database**. These database types are used for fast annotation with reference sequences and do not allow for metadata. If you require annotation with metadata, for instance when using a RNAcentral database with GO terms in order to build a functional profile, this option can not be used. Instead, the sequence list option must be used, even though it is slightly slower.

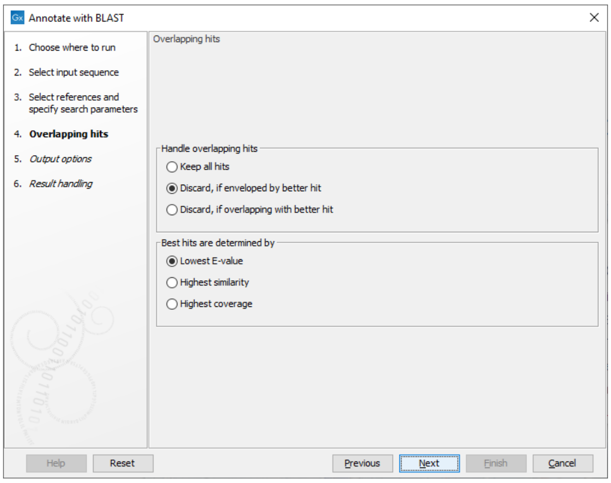
The search parameters can be modified using the following settings:

- **Genetic code**. The genetic code used when translating the nucleotide sequences before searching against the protein references.
- **Maximum E-value**. Maximum expectation value (E-value) threshold for accepting hits.
- **Minimum identity (%)**. The minimum percent identity for a hit to be accepted. The percent identity is calculated based on the number of amino acid matches when using protein reference sequences (blastx), and based on the number of nucleotide matches when using nucleotide reference sequences (blastn). Notice. when annotating with a Protein sequence list of clustered sequences such as UniRef50, this should be lowered depending on the level of clustering in the database.
- **Minimum reference sequence coverage (%)**. The minimum length fraction of the reference sequence that must be matched. Notice: this is length fraction per hit (HSP), and should be kept low when searching for non-contiguous matches.

Adjustment can be made to the annotation hits by the following setting:

- **CDS adjustment.** The found annotation hits will be adjusted to begin with a start codon, end with a stop codon and not contain any stop codons in between. The adjustment can extend the annotation to up to 110 percent of the length of the reference gene and will not be shorter than 90 percent of the reference gene length. The frame of the translation may change from the original alignment.

The next step (figure 11.3), determines how to handle when multiple overlapping hits are found on the input query sequence.



Annotate with BLAST

1. Choose where to run
2. Select input sequence
3. Select references and specify search parameters
4. **Overlapping hits**
5. *Output options*
6. *Result handling*

Overlapping hits

Handle overlapping hits

○ Keep all hits
● Discard, if enveloped by better hit
○ Discard, if overlapping with better hit

Best hits are determined by

● Lowest E-value
○ Highest similarity
○ Highest coverage

Figure 11.3: *Settings for handling overlapping hits*

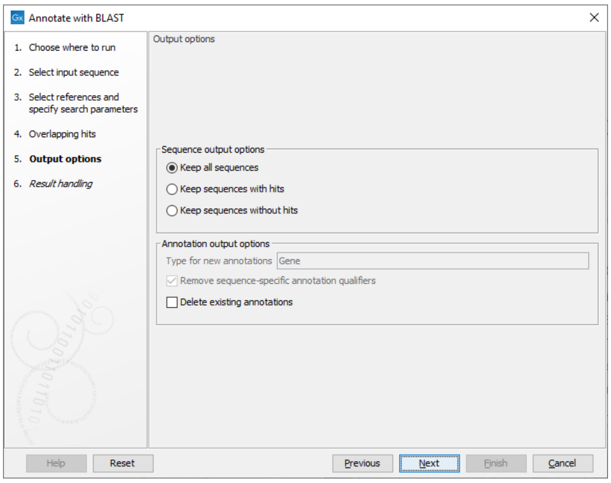
The following options are available:

- **Keep all hits.** all hits that meet the search criteria are annotated on the input query sequence.
- **Discard, if enveloped by better hit.** If a hit covers the same region or part of the same region as a better hit, it is discarded.
- **Discard, if overlapping with better hit.** If a hit overlaps the same region as a better hit, it is discarded.

Best hits are determined by:

- **Lowest E-value.** hits with the lowest E-value are kept. Ties are resolved by highest similarity, subsequently highest coverage.
- **Highest similarity.** hits with the highest similarity are kept. Ties are resolved by lowest E-value, subsequently highest coverage.
- **Highest coverage.** hits with the highest coverage are kept. Ties are resolved by lowest E-value, subsequently highest similarity.

The output options step (figure 11.4), has the following options:



Output options

Sequence output options

● Keep all sequences

○ Keep sequences with hits

○ Keep sequences without hits

Annotation output options

Type for new annotations: Gene

☑ Remove sequence-specific annotation qualifiers

□ Delete existing annotations

Figure 11.4: *Specifying output options*

- **Type for new annotations**. When using a protein database as source, all new annotations will be of type 'CDS'. However, when using a nucleotide sequence list, or a nucleotide sequence BLAST database, there is no general annotation type to apply. The default output annotation type will be 'Gene', but this can be customized if necessary.

- **Remove sequence-specific annotation qualifiers**. Annotation qualifiers such as 'translation' and 'codon_start' may no longer be accurate on the new annotations. This option removes such qualifiers.

- **Delete existing annotations**. Existing annotations on the input sequences will not be copied to the output sequences.

The following sequence output options are available:

- **Keep all sequences**

- **Keep sequences with hits**. This option can be useful for filtering input sequences for certain regions.

- **Keep sequences without hits**. This option can be useful for comparing sequence lists.

The final step controls which outputs are created. Notice, that reports can be aggregated using the Combine Reports tool.

11.3 Annotate with DIAMOND

The **Annotate with DIAMOND** tool allows you to annotate a DNA sequence using a set of known protein reference sequences. This tool can be used on sequences without any pre-existing annotations: it is not necessary to annotate the DNA sequences with genes or coding regions. For more information about the DIAMOND aligner, see section 11.5.

The tools can be used for various purposes, e.g. transferring annotations from a known reference, annotate the presence of AMR or virulence markers in a genome, or to filter contigs or sequences based on the presence of a set of genes.

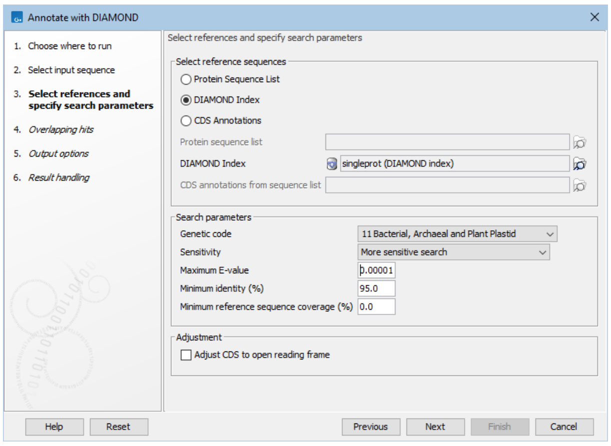
For annotating DNA sequences from a set of non-coding reference sequences, the **Annotate with BLAST** tool may be used instead. However, the **Annotate with DIAMOND** tool is in general the fastest option when working with coding regions.

If the input sequences are already annotated with CDS annotations, it is also possible to use the **Annotate CDS with Best BLAST Hit** and **Annotate CDS with Best DIAMOND Hit** tools - see section 11.4 for more information.

To start the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Annotate with DIAMOND** ( )

The first wizard step (figure 11.5), specifies the reference and search parameters.



Select references and specify search parameters

Select reference sequences

○ Protein Sequence List

● DIAMOND Index

○ CDS Annotations

Protein sequence list:

DIAMOND Index: singleprot (DIAMOND index)

CDS annotations from sequence list:

Search parameters

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Sensitivity: More sensitive search

Maximum E-value: 0.00001

Minimum identity (%): 95.0

Minimum reference sequence coverage (%): 0.0

Adjustment

□ Adjust CDS to open reading frame

Figure 11.5: *Selecting references and specifying search parameters.*

The following sources can be used to annotate the input sequences:

- **Protein Sequence List**. The nucleotide input query will be searched against the sequences in the protein sequence list. The nucleotide input will be translated using the chosen genetic code. If the reference protein sequence list contains metadata, this metadata will be transferred to the resulting annotations on the input query sequence.
- **DIAMOND Index**. DIAMOND indexes can be created using the Create DIAMOND Index tool. This works similar to the Protein Sequence List option, but can be faster since the index can be reused. When using the DIAMOND index, the name and description of detected reference sequences are transferred to the input query sequence.
- **CDS Annotations**. This option uses a nucleotide sequence source with existing annotations as a source. All CDS annotations are extracted and translated to a protein database, which is searched similar to the previous options. All qualifiers on the detected source annotations are transferred to the input query sequence.

As can be seen above, metadata (such as GO terms and taxonomy information) is handled differently depending on the database source:

- **Protein sequence list**. Sequence lists may contain metadata, which can be inspected in the table view of the sequence list. Such metadata is transferred to the annotations created by this tool.

- **DIAMOND Index**. If a DIAMOND index was created from a protein sequence list containing metadata, the original metadata will be transferred to the annotations created by this tool.
- **CDS annotations from sequence list**. Annotations are transferred together with any metadata qualifiers the annotations contain.

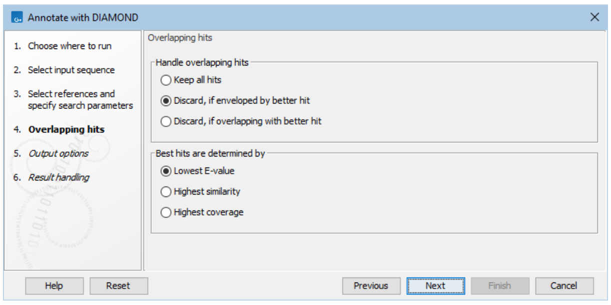
The search parameters can be modified using the following settings:

- **Genetic code**. The code used when translating the nucleotide sequences before searching against the protein references.
- **Sensitivity**. Select DIAMOND sensitivity setting.
- **Maximum E-value**. Maximum expectation value (E-value) threshold for saving hits.
- **Minimum identity (%)**. The minimum percent amino acid identity for a hit to be accepted. Notice: when annotating with a Protein sequence list of clustered sequences such as UniRef50, this should be lowered depending on the level of clustering in the database.
- **Minimum reference sequence coverage (%)**. The minimum length fraction of the reference sequence that must be matched. Notice: this is length fraction per hit (HSP), and should be kept low when searching for non-contiguous matches.

Adjustment can be made to the annotation hits by the following setting:

- **CDS adjustment**: The found annotation hits will be adjusted to begin with a start codon, end with a stop codon and not contain any stop codons in between. The adjustment can extend the annotation to up to 110 percent of the length of the reference gene and will not be shorter than 90 percent of the reference gene length. The frame of the translation may change from the original alignment.

The next step (figure 11.6), determines how to handle when multiple overlapping hits are found on the input query sequence.



Overlapping hits

Handle overlapping hits
○ Keep all hits
● Discard, if enveloped by better hit
○ Discard, if overlapping with better hit

Best hits are determined by
● Lowest E-value
○ Highest similarity
○ Highest coverage

Figure 11.6: *Settings for handling overlapping hits.*

The following options are available:

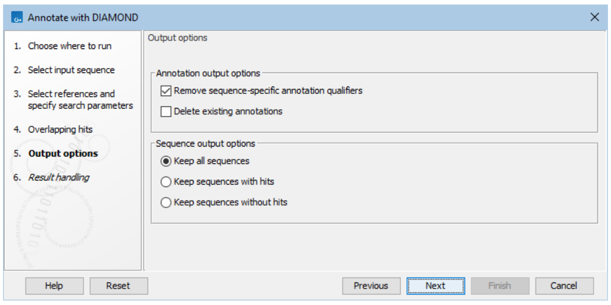
- **Keep all hits**: all hits that meet the search criteria are annotated on the input query sequence.

- **Discard, if enveloped by better hit**: If a hit covers the same region or part of the same region as a better hit, it is discarded.
- **Discard, if overlapping with better hit**: If a hit overlaps the same region as a better hit, it is discarded.

Best hits are determined by:

- **Lowest E-value**: hits with the lowest E-value are kept. Ties are resolved by highest similarity, subsequently highest coverage.
- **Highest similarity**: hits with the highest similarity are kept. Ties are resolved by lowest E-value, subsequently highest coverage.
- **Highest coverage**: hits with the highest coverage are kept. Ties are resolved by lowest E-value, subsequently highest similarity.

The output options step (figure 11.7), has the following options:



Annotate with DIAMOND

1. Choose where to run
2. Select input sequence
3. Select references and specify search parameters
4. Overlapping hits
5. **Output options**
6. *Result handling*

Output options

Annotation output options

☑ Remove sequence-specific annotation qualifiers

□ Delete existing annotations

Sequence output options

● Keep all sequences

○ Keep sequences with hits

○ Keep sequences without hits

Figure 11.7: *Specifying output options.*

- **Remove sequence-specific annotation qualifiers**. Annotation qualifiers such as 'translation' and 'codon_start' may no longer be accurate on the new annotations. This option removes such qualifiers.
- **Delete existing annotations**. Existing annotations will not be copied to the output sequences.

The following sequence output options are available:

- **Keep all sequences**
- **Keep sequences with hits**. This option can be useful for filtering input sequences for certain regions.
- **Keep sequences without hits**. This option can be useful for comparing sequence lists.

The final step controls which outputs are created. Notice, that reports can be aggregated using the Combine Reports tool.

11.4 Annotate CDS with Best BLAST Hit

The **Annotate CDS with Best BLAST Hit** tool will allow you to annotate a set of contigs containing CDS annotations with their best BLAST hit.

To start the analysis, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Annotate CDS with Best BLAST Hit** ( )

Several parameters are available:

- **Genetic code**. The genetic code used for translating CDS to proteins.
- **BLAST database**. A protein BLAST database. Popular BLAST protein databases can be downloaded using the Download BLAST Database tool or created using a the Create BLAST Database tool.
- **Maximum E-value**. Maximum expectation value (E-value) threshold for saving hits.

Metadata from the sequences used to create the BLAST database (such as GO terms or taxonomy information) will not be transferred by this tool. If metadata is relevant, consider using the **Annotate CDS with Best DIAMOND Hit** tool instead.

Note that choosing a very large BLAST database with millions of sequences (e.g. the nt, nr and refeseq_protein databases from the NCBI) will slow down the algorithm considerably, especially when there are many CDS in the input. Therefore, we recommend to use a medium-sized database such as "swissprot". In the wizard, you can choose between databases stored locally ( ) or remotely on the server ( ). If you create a workflow that you plan to run on a server, you should avoid locking the BLAST database parameter as the chosen database may not exist on the server.

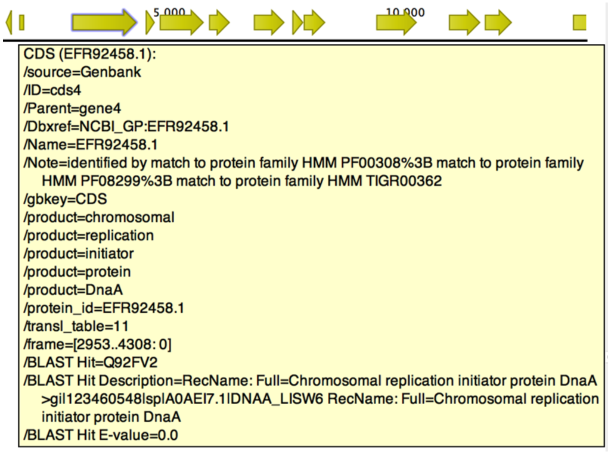
If you select **Create Report**, the tool will create a summary report table. The report is divided in three parts:

- **Input**. Contains information about the size of the contigs and CDS used as input.
- **BLAST database**. The protein BLAST database used in the search, together with its description, location, and size.
- **Output**. The total number (and percent) of CDS that were annotated with their best BLAST hit.

The tool will output a copy of the input file containing the following fields when a hit for a CDS is found (figure 11.8):

- **BLAST Hit**. Accession number of the best BLAST Hit in the BLAST database.
- **BLAST Hit Description**. Description of the matching protein, as present in the BLAST database.
- **BLAST Hit E-value**. The E-value of the match.

The tool can also output an annotation table summarizing information about the annotations added to the sequence list.



```
CDS (EFR92458.1):
/source=Genbank
/ID=cds4
/Parent=gene4
/Dbxref=NCBI_GP:EFR92458.1
/Name=EFR92458.1
/Note=identified by match to protein family HMM PF00308%3B match to protein family
   HMM PF08299%3B match to protein family HMM TIGR00362
/gbkey=CDS
/product=chromosomal
/product=replication
/product=initiator
/product=protein
/product=DnaA
/protein_id=EFR92458.1
/transl_table=11
/frame=[2953..4308: 0]
/BLAST Hit=Q92FV2
/BLAST Hit Description=RecName: Full=Chromosomal replication initiator protein DnaA
   >gi|123460548|sp|A0AEI7.1|DNAA_LISW6 RecName: Full=Chromosomal replication
   initiator protein DnaA
/BLAST Hit E-value=0.0
```

Figure 11.8: *BLAST Best Hit annotations added to gene cds4 of h. pylori.*

11.5 Annotate CDS with Best DIAMOND Hit

Annotate CDS with Best DIAMOND Hit allows you to annotate a set of contigs containing CDS annotations with their best DIAMOND hit. This tool is particularly useful for large data sets, as an alternative to Annotate CDS with Best BLAST Hit.

DIAMOND is a sequence aligner for protein and translated DNA searches, designed for high performance analysis of big sequence data, see https://github.com/bbuchfink/diamond. The key features are:

- Pairwise alignment of proteins and translated DNA at 500x-20,000x speed of BLAST.
- Frameshift alignments for long read analysis.
- Low resource requirements and suitable for running on standard desktops or laptops.

To start the analysis, go to:

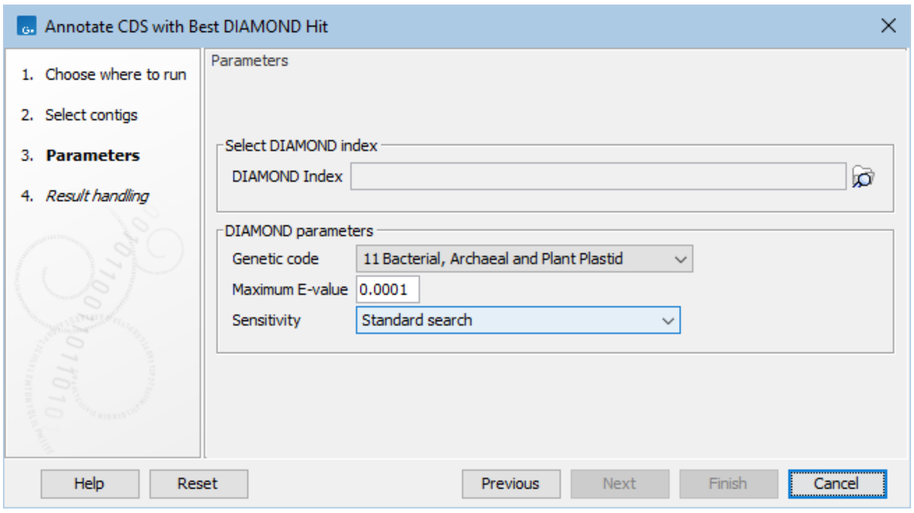
Toolbox | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Annotate CDS with Best DIAMOND Hit** ( )

Select the CDS-annotated contigs to be annotated with DIAMOND hits.

In the **Parameters** dialog page (figure 11.9), set the following

- **DIAMOND Index**. Select the relevant indexes.

  Indexes can be generated by downloading a database with the Download Protein Database tool (section 16.1) and building and index using the Create DIAMOND Index tool (section 16.4).
- **DIAMOND parameters**
  - **Genetic code**. The genetic code used for translating CDS to proteins.
  - **Maximum E-value**. Maximum expectation value (E-value) threshold for saving hits.
  - **Sensitivity**. Select DIAMOND sensitivity setting.



Parameters

Select DIAMOND index

DIAMOND Index

DIAMOND parameters

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Maximum E-value: 0.0001

Sensitivity: Standard search

Figure 11.9: *Annotate CDS with Best DIAMOND Hit parameters.*

The tool will output a copy of the input file with the DIAMOND Hit annotations. The tool can also output an annotation table summarizing information about the annotations added to the sequence list. Finally it is possible to generate a report containing information about the input file, the DIAMOND database and the amount of CDS annotated with a DIAMOND hit.

If a DIAMOND index was created from a protein sequence list containing metadata (such as GO terms or taxonomy information), the original metadata will be transferred to the annotations created by this tool.

11.6 Annotate CDS with Pfam Domains

The **Annotate CDS with Pfam Domains** tool will allow you to annotate a set of contigs containing CDS annotations with Pfam and GO terms. To start the analysis, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Annotate CDS with Pfam Domains** ( )

The following parameters are available:

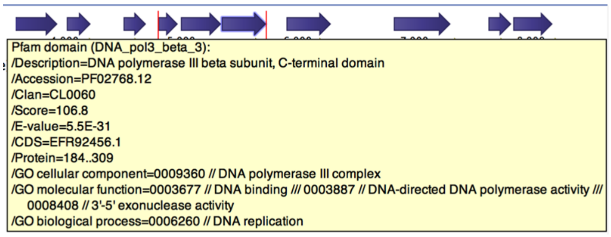
- **Genetic code.** The genetic code used for translating CDS to proteins.
- **Pfam database.** The Pfam database. This database can be downloaded using the "Download Pfam Database" tool.
- **Use profile's gathering cutoffs.** Use cutoffs specifically assigned to each family by the curator instead of manually assigning the Significance cutoff.
- **Significance cutoff.** The E-value (expectation value) describes the number of hits one would expect to see by chance when searching a database of a particular size.
- **Remove overlapping matches from the same clan.** Perform post-processing of the results where overlaps between hits are resolved by keeping the hit with the smallest e-value.

- **GO database**. The GO database, used to map between Pfam domains and GO terms. The GO database can be downloaded using the Download GO Database tool ((see section 16.2). If the database is not specified, no GO annotation will be added.
- **GO subset**. A subset of the GO database. Since many GO terms are too general or too specific, several meaningful subsets of GO terms are provided. See `https://geneontology.org/docs/download-ontology/`.

If you select **Create report**, the tool will create a summary report table. The report is divided in three parts

- **Input**. Contains information about the size of the contigs and CDS used as input.
- **Output**. The total number (and percent) of CDS that were annotated with a Pfam domain or a GO term, as well as the total number of Pfam domains and GO terms added.
- **Pfam database**.The Pfam database used in the search together with its version and size.
- **GO database**. The GO database (or subset) used in the search together with its version, size, and the number of Pfam domains mapping to at least one term.

The tool will output a copy of the input file containing Pfam annotations when a Pfam domain was found in a CDS, as shown in figure 11.10. The annotation contains the following fields:



```
Pfam domain (DNA_pol3_beta_3):
/Description=DNA polymerase III beta subunit, C-terminal domain
/Accession=PF02768.12
/Clan=CL0060
/Score=106.8
/E-value=5.5E-31
/CDS=EFR92456.1
/Protein=184..309
/GO cellular component=0009360 // DNA polymerase III complex
/GO molecular function=0003677 // DNA binding /// 0003887 // DNA-directed DNA polymerase activity ///
    0008408 // 3'-5' exonuclease activity
/GO biological process=0006260 // DNA replication
```

Figure 11.10: *Pfam and GO annotations added to gene cds4 of h. pylori.*

- **Description**. A description of the Pfam domain.
- **Accession**. The accession number of the Pfam domain.
- **Clan**. The clan that the domain belong to (if any).
- **Score**. The score
- **E-value**. The E-value of the match.
- **CDS**. The CDS that contains this domain.
- **Protein**. The protein region (in aa coordinates) that encodes for the domain.
- **GO cellular component**. GO terms of the cellular component domain which are related to the Pfam domain.
- **GO molecular function**. GO terms of the molecular function domain which are related to the Pfam domain.

- **GO biological process**. GO terms of the biological process domain which are related to the Pfam domain.

The tool can also output an annotation table summarizing information about the annotations added to the sequence list.

11.7 Build Functional Profile

To compute the number of reads in a sample mapping to regions involved with Pfam domains, or BLAST or DIAMOND hits, you can run the Build Functional Profile tool by going to:

> **Toolbox** | **Microbial Genomics Module** () | **Functional Analysis** () | **Build Functional Profile** ()

In the first wizard (figure 11.11), select the read mapping for which you want to build the functional profile.

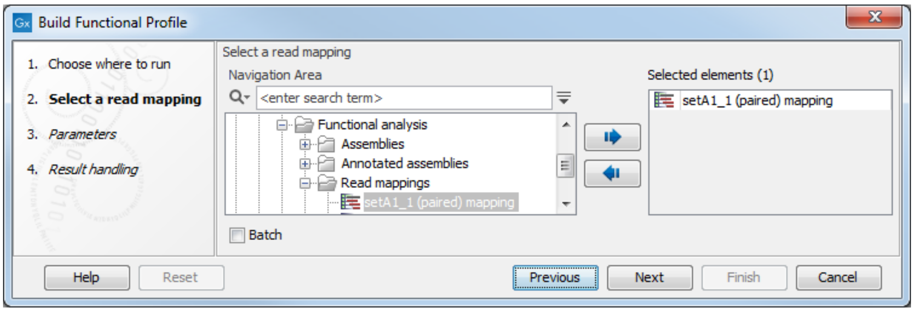


Figure 11.11: *Select a read mapping.*

The parameters that can be set are seen in figure 11.12:

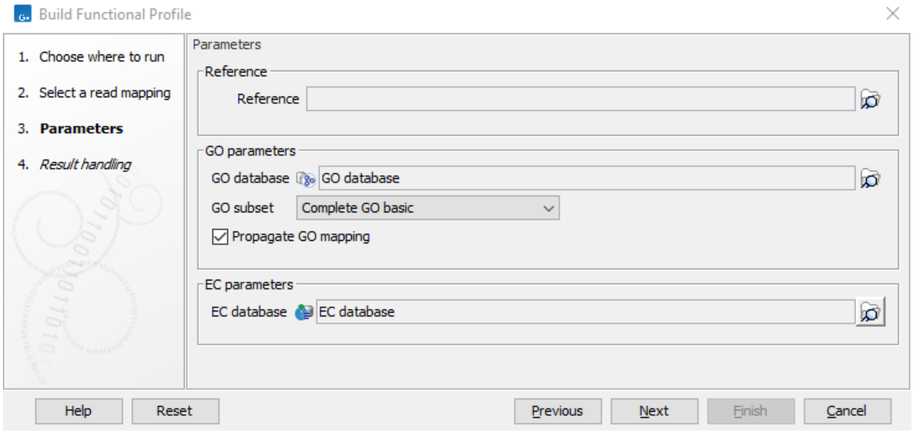


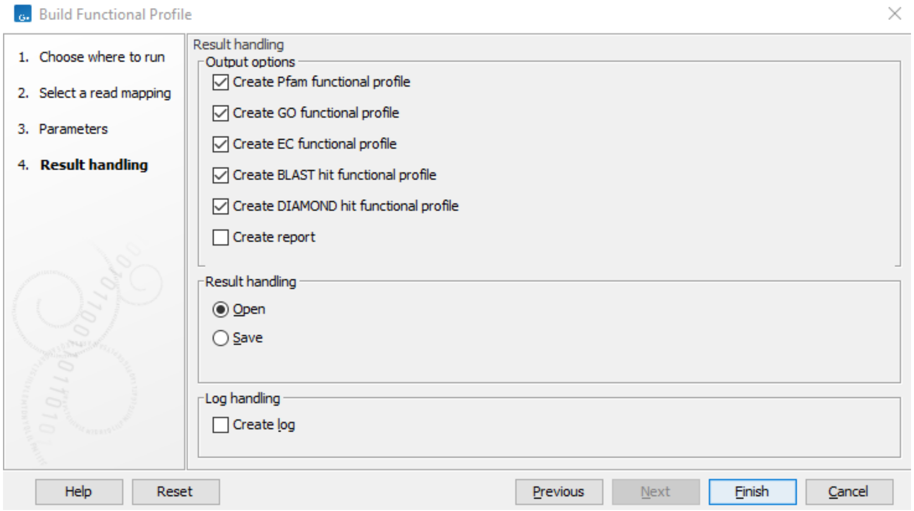
Figure 11.12: *Specify a reference, a GO database and an EC database.*

- **Reference**. A reference set of contigs annotated with Pfam domains, BLAST, or DIAMOND hits. If the read mapping contains an annotated genome, this parameter is optional.
- **GO database**. The GO database. If the reference contains Pfam domains, this database can be used to map from Pfam domains to GO terms. If the BLAST or DIAMOND hits contain GO-terms annotations, this will be also matched against the database and appear in the

GO abundance table output. The GO database can be downloaded using the Download Ontology Database tool (see section 16.2).

- **GO subset**. A subset of the GO database. Since many GO terms are too general or too specific, several meaningful subsets of GO terms are provided. See https://geneontology.org/docs/download-ontology/.
- **Propagate GO mapping**. When selected, GO terms are mapped to all their ancestor terms. For example, the Pfam domain "CutC" maps to the GO term "0005507 // copper ion binding". If **Propagate GO mapping** is enabled, the tool would also map to more general GO terms such as "0055070 // copper ion homeostasis", "0055076 // transition metal ion homeostasis", and "0065007 // biological regulation".
- **EC database**. The EC database. If the reference contains BLAST or DIAMOND domains, this database can be used to map EC terms. The EC database can be downloaded using the Download Ontology Database (see section 16.2).

You can then select which output elements should be generated figure 11.13.



Result handling

Output options

☑ Create Pfam functional profile

☑ Create GO functional profile

☑ Create EC functional profile

☑ Create BLAST hit functional profile

☑ Create DIAMOND hit functional profile

□ Create report

Result handling

● Open

○ Save

Log handling

□ Create log

Figure 11.13: *Specify what type of output you want the tool to generate.*

- **Create Pfam functional profile**. Abundance table obtained by counting reads overlapping Pfam domains.
- **Create GO functional profile**. Abundance table obtained by counting reads overlapping GO terms. Note that a database must be specified in order to build a GO functional profile. For BLAST and DIAMOND hits, the GO-terms specified on the annotations are used, but preexisting GO annotations on pfam domains are ignored by this tool.
- **Create EC functional profile**. Abundance table obtained by counting reads overlapping EC terms. Note that a database must be specified in order to build an EC functional profile.
- **Create BLAST hit functional profile**. Abundance table obtained by counting reads overlapping BLAST hits.
- **Create DIAMOND hit functional profile**. Abundance table obtained by counting reads overlapping DIAMOND hits.

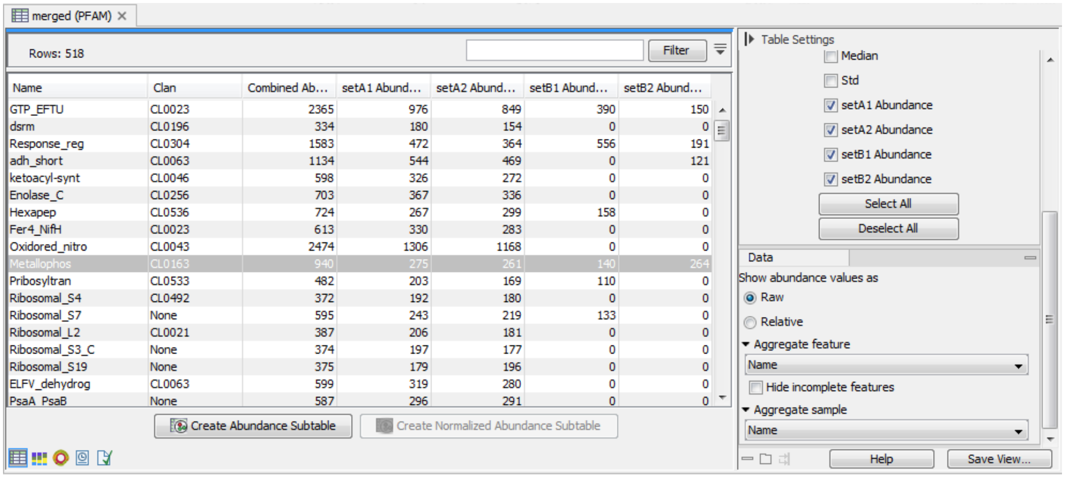
- **Create Report**. A report stating statistics about the input reference contigs and read mapping, as well as the number of matches to each feature.

The resulting functional abundance tables store the number of reads corresponding to each Pfam domain, GO term, EC number, best BLAST hit or best DIAMOND hit.

11.7.1 Functional profile abundance table

The functional profile abundance table displays the names of the function, along with their clan, a combined abundance. The table can be visualized using the Stacked bar charts and Stacked area charts function, as well as the Sunburst charts.

- **Table view** (▦) (figure 11.14)



merged (PFAM)

Rows: 518

Name	Clan	Combined Ab...	setA1 Abund...	setA2 Abund...	setB1 Abund...	setB2 Abund...
GTP_EFTU	CL0023	2365	976	849	390	150
dsrm	CL0196	334	180	154	0	0
Response_reg	CL0304	1583	472	364	556	191
adh_short	CL0063	1134	544	469	0	121
ketoacyl-synt	CL0046	598	326	272	0	0
Enolase_C	CL0256	703	367	336	0	0
Hexapep	CL0536	724	267	299	158	0
Fer4_NifH	CL0023	613	330	283	0	0
Oxidored_nitro	CL0043	2474	1306	1168	0	0
Metallophos	CL0163	940	275	261	140	264
Pribosyltran	CL0533	482	203	169	110	0
Ribosomal_S4	CL0492	372	192	180	0	0
Ribosomal_S7	None	595	243	219	133	0
Ribosomal_L2	CL0021	387	206	181	0	0
Ribosomal_S3_C	None	374	197	177	0	0
Ribosomal_S19	None	375	179	196	0	0
ELFV_dehydrog	CL0063	599	319	280	0	0
PsaA_PsaB	None	587	296	291	0	0

Table Settings

- ☐ Median
- ☐ Std
- ☑ setA1 Abundance
- ☑ setA2 Abundance
- ☑ setB1 Abundance
- ☑ setB2 Abundance

Data

Show abundance values as

- ● Raw
- ○ Relative

Aggregate feature: Name

☐ Hide incomplete features

Aggregate sample: Name

Filter
Create Abundance Subtable
Create Normalized Abundance Subtable
Select All
Deselect All
Help
Save View...

Figure 11.14: *Functional profile abundance table.*

The table displays the following columns:

- **ID**: internal ID the abundance tables use for ordering the samples. IDs are unique, while Names are not necessarily, so that when merging abundance tables taxa with the same ID will be combined.
- **Name**: the name of the taxon, specified by the reference database or the NCBI taxonomy. If the name contains the text "(Unknown)", it indicates that this taxon corresponds to a higher-level node in the taxonomy, and that this node had a significant amount of reads associated to ancestor taxons that are present in the database but were disqualified. This indicates that there was some organism in the sample for which there is no exactly matching reference in the database, but is most likely closely related to this taxon. If the name does not contain the text "(Unknown)", it means that the sample contains this exact taxon, which is present in the database.
- **Clan**: a collection of related Pfam entries. The relationship may be defined by similarity of sequence, structure or profile-HMM.
- **Combined Abundance**: total number of reads for the function across all samples
- **Min**, **Max**, **Mean**, **Median** and **Std**: respectively minimum, maximum, mean, median and standard deviation of the number of reads for the fucntion across all samples
- **Abundance for the sample**: number of reads for each sample

In the right side panel, under the tab Data, you can switch between raw and relative abundances (relative abundances are computed as the ratio between the coverage for a function in a specific sample and the amount of coverage in the sample). You can also combine absolute counts and relative abundances by selecting the Clan level in the **Aggregate feature** drop-down menu.

Finally, if you have previously annotated your table with Metadata (see section 6.8), you can **Aggregate sample** by the groups previously defined in your metadata table. This is useful when for example analyzing replicates from the same sample origin.

Under the table, the following actions are available:

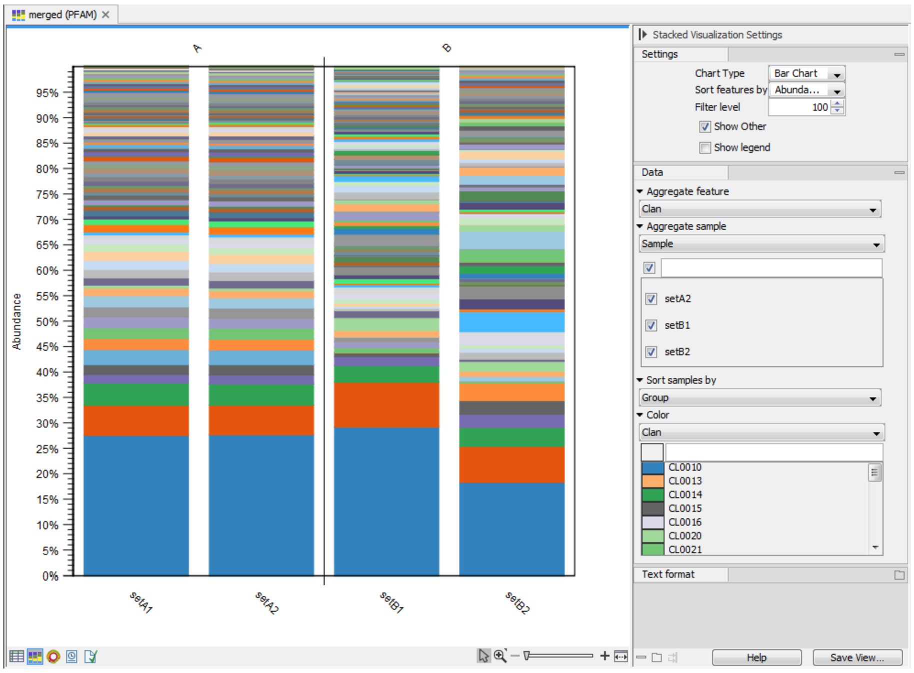
  - **Create Abundance Subtable** will create a table containing only the selected rows.
  - **Create Normalized Abundance Subtable** will create a table with all rows normalized on the values of a single selected row. The row used for normalization will disappear from the new abundance table. The normalization scales the abundance table linearly, where the scaling factor is calculated by determining the average abundance across all samples and for each sample scale it to the average for the reference. Note that to be enabled, the selected row for normalization can only have non null abundance values. If you have zero values in some samples for the control, you will need to generate a new abundance table where these samples are not present. If the abundance table is obtained from merging single-sample abundance tables, then the merge should be redone excluding the samples with zero control read counts.

- **Stacked Bar Chart and Stacked Area Chart** (▦)

  Choose which chart you want to see using the drop down menu in the upper right corner of the side panel. In the Stacked Bar (figure 11.15) and Stacked Area Charts (figure 11.16), the metadata can be used to aggregate groups of columns (samples) by selecting the relevant metadata category in the right hand side panel. Also, the data can be aggregated at any taxonomy level selected. The relevant data points will automatically be summed accordingly.

  Holding the pointer over a colored area in any of the plots will result in the display of the corresponding taxonomy label and counts. **Filter level** allows to modify the number of features to be shown in the plot. For example, setting the value to 10 means that the 10 most abundant features of each sample will be shown in all columns. The remaining features are grouped into "Other", and will be shown if the option is selected in the right hand side panel. One can select which taxonomy level to color, and change the default colors manually. Colors can be specified at the same taxonomy level as the one used to aggregate the data or at a lower level. When lower taxonomy levels are chosen in the data aggregation field, the color will be inherited in alternating shadings. It is also possible to sort samples by metadata attributes, and to show groups of samples without collapsing their stacks, as well as change the label of each stack or group of stacks. Features can be sorted by "abundance" or "name" using the drop down menu in the right hand side panel. Using the bottom right-most button (**Save/restore settings** (☰)), the settings can be saved and applied in other plots, allowing visual comparisons across analyses.

- **Zoomable Sunbursts** (◎) The Zoomable Sunburst viewer lets the user select how many taxonomy level counts to display, and which level to color. Lower levels will inherit the color in alternating shadings. Taxonomy and relative abundances (the ratio between the coverage of the species in a specific sample and the total amount of coverage in the sample) are



Stacked Visualization Settings

Settings

Chart Type: Bar Chart

Sort features by: Abunda...

Filter level: 100

☑ Show Other

□ Show legend

Data

Aggregate feature: Clan

Aggregate sample: Sample

☑ setA2

☑ setB1

☑ setB2

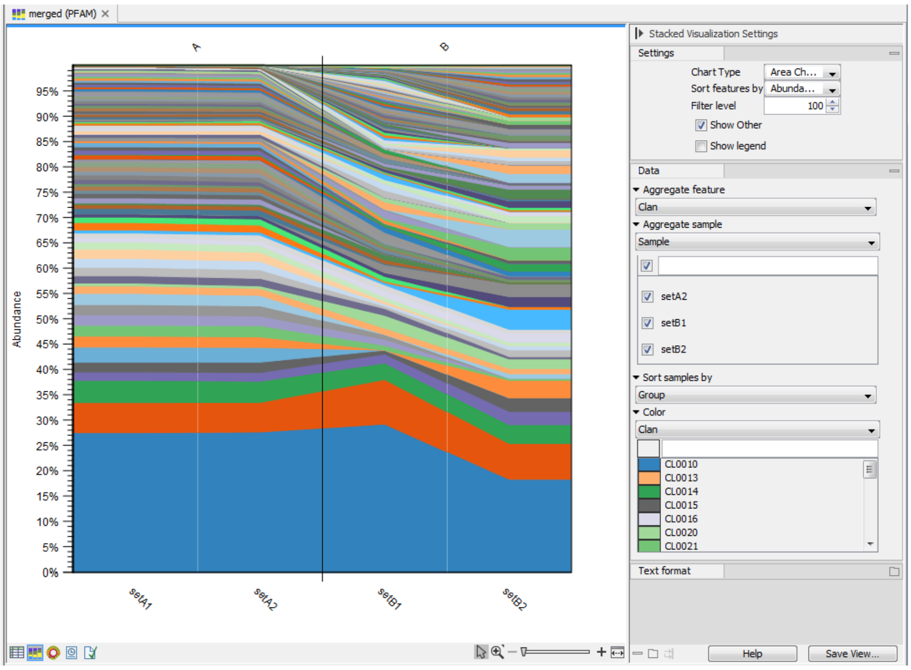
Sort samples by: Group

Color: Clan

🟦	CL0010
🟧	CL0013
🟩	CL0014
⬛	CL0015
⬜	CL0016
🟩	CL0020
🟩	CL0021

Text format

Figure 11.15: *Stacked bar chart.*



Stacked Visualization Settings

Settings

Chart Type: Area Ch...

Sort features by: Abunda...

Filter level: 100

☑ Show Other

□ Show legend

Data

Aggregate feature: Clan

Aggregate sample: Sample

☑ setA2

☑ setB1

☑ setB2

Sort samples by: Group

Color: Clan

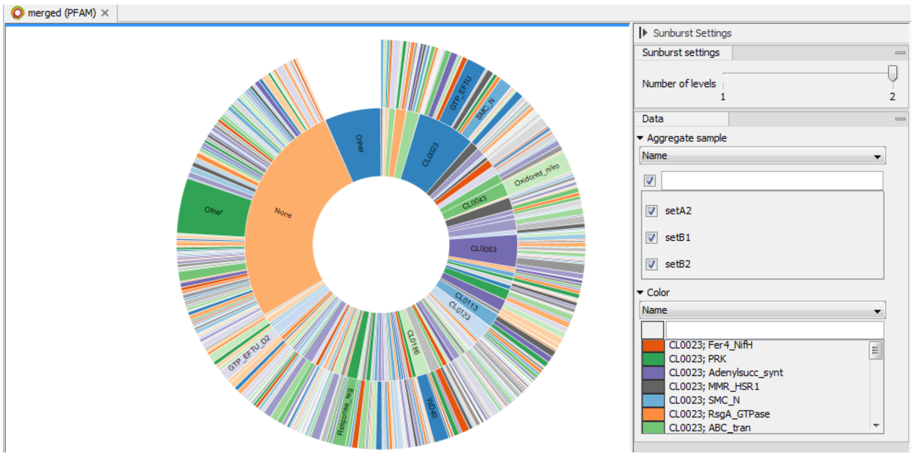
🟦	CL0010
🟧	CL0013
🟩	CL0014
⬛	CL0015
⬜	CL0016
🟩	CL0020
🟩	CL0021

Text format

Figure 11.16: *Stacked area chart.*

displayed in a legend to the left of the plot when hovering over the sunburst viewer with the mouse. The metadata can be used to select which sample or group of samples to show in the sunburst (figure 11.17).

Clicking on a lower level field will render that field the center of the plot and display lower level counts in a radial view. Clicking on the center field will render the level above the current view the center of the view.



Sunburst Settings

Sunburst settings

Number of levels: 1 – 2

Data

Aggregate sample: Name

☑

☑ setA2

☑ setB1

☑ setB2

Color: Name

🟧 CL0023; Fer4_NifH
🟩 CL0023; PRK
🟪 CL0023; Adenylsucc_synt
⬛ CL0023; MMR_HSR1
🟦 CL0023; SMC_N
🟧 CL0023; RsgA_GTPase
🟩 CL0023; ABC_tran

Figure 11.17: *Sunburst view.*

11.8 Infer Functional Profile

This tool is currently in beta. Feedback on this plugin is welcome - please get in touch by email (ts-bioinformatics@qiagen.com) and let us know how it can be improved.

For OTU abundance tables it is possible to infer an approximate functional profile using the Infer Functional Profile (beta) tool. In order to run this tool you need a PICRUSt2 Multiplication Table [Douglas et al., 2020] which may be imported using the Import PICRUSt2 Multiplication Table (beta) 16.6 and optionally an EC database that may be downloaded using the Download Ontology Database 16.6 tool.

To infer a functional profile from an OTU table, go to

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Infer Functional Profile (beta)** ( )

In the first wizard (figure 11.18), select the OTU table for which you want to build the functional profile. Note that the OTU table must have OTU sequences.

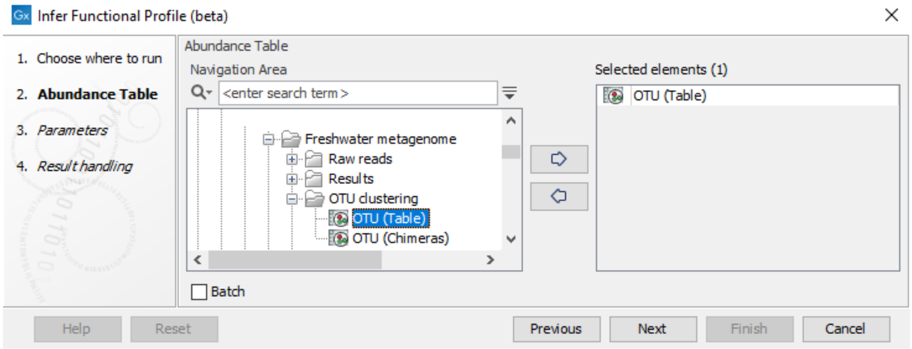
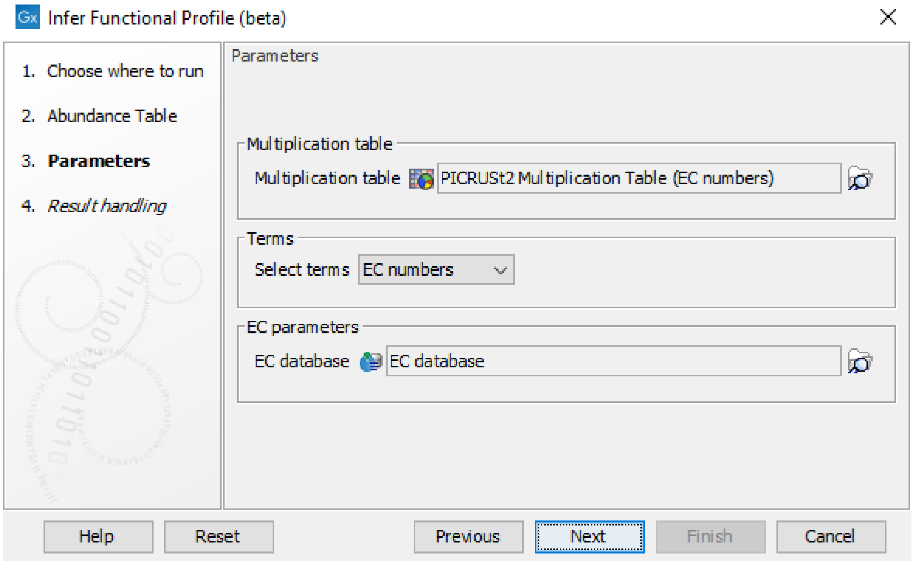


Figure 11.18: *Select an OTU abundance table.*

In the second step of the wizard (figure 11.19) the terms for which to produce a functional abundance profile can be selected. Note that when selecting to create an EC abundance profile, an additional EC database is required.

The resulting functional abundance tables store the inferred number of reads corresponding to each of the selected terms in a separate table.



Infer Functional Profile (beta)

1. Choose where to run
2. Abundance Table
3. **Parameters**
4. *Result handling*

Parameters

Multiplication table

Multiplication table: PICRUSt2 Multiplication Table (EC numbers)

Terms

Select terms: EC numbers

EC parameters

EC database: EC database

Figure 11.19: *Specify the functional terms for which a functional abundance table shall be inferred.*

Inference of functional abundances from 16S/ITS data PICRUSt2 Multiplication Tables can be imported using the Import PICRUSt2 Multiplication Table (beta) tool. The multiplication tables contain kmer frequency profiles, associated rRNA copy numbers and term multipliers, i.e. how often a certain functional term is encountered on the genomic sequence of the associated rRNA sequence. The Infer Functional Profile (beta) algorithm works by comparing kmer frequency profiles for each identified OTU with the stored kmer frequency profiles in a PICRUSt2 Multiplication Table to find the nearest neighbor to the OTU under consideration in the Multiplication Table. For this nearest neighbor, both the rRNA copy number and term multiplication numbers are available. From a single OTU the predicted term multiplicity is obtained by dividing the read count for the OTU by the identified rRNA copy number and multiplying it by the identified term multiplication number. To obtain the final inferred term read count, the individual predictions for all OTUs are summed up per term.

The Infer Functional Profile (beta) algorithm is inspired by two published methods PICRUSt2 [Douglas et al., 2020] and Piphillin [Narayan et al., 2020]. Note that PICRUSt2 [Douglas et al., 2020] and Piphillin [Narayan et al., 2020] do not use kmer frequency profiles but alignments (and for PICRUSt2 optimal tree positioning of a reference) for the identification of the nearest neighbor(s), which typically have a higher precision but are also slower to compute. Typically, it is not expected that high precision is required for the identification of the nearest neighbors as most OTUs are most likely not represented exactly in the database and a close neighbor is typically good enough. While this is true for well-represented species, it has been shown in [Douglas et al., 2020] that only a single nearest neighbor may be a bad predictor for the rRNA and term copy numbers. In this respect we expect the Infer Functional Profile (beta) tool to be comparable to Piphillin [Narayan et al., 2020].

11.9 Identify Pathways

The Identify Pathways tool takes a functional abundance table with EC terms or a differential abundance table with EC terms as input and translates these into pathway calls using a pathway database. A pathway database can be obtained with the Download Pathway Database tool, see section 16.3. If the input is an abundance table, the called pathways will correspond to all pathways present in the sample. If the input is a differential abundance table, the called pathways are the pathways that have been up or down regulated between two groups of samples.

The algorithm produces a range of solutions for the pathway calls:

- The **naive solution** where a pathway is called if it contains at least one of the functional terms present in the input table.
- The **minimum solution** where the smallest set of pathways is chosen such that all terms from the input are present in at least one of the chosen pathways. This is similar to MinPath [Ye and Doak, 2009], only that the algorithm used is based on a greedy minimum set cover strategy and thus only finds an approximate solution to the stated minimization problem.
- The **confidence** based solution where each pathway is associated with a confidence call based on randomized evaluations of the minimum set cover strategy. In this way, each pathway call is be associated with a confidence for the presence of that pathway. The naive solution and the minimum solution are the outer goal posts, whereas the confidence based solution gives a smooth metric in-between.

To run the Identify Pathways tool go to

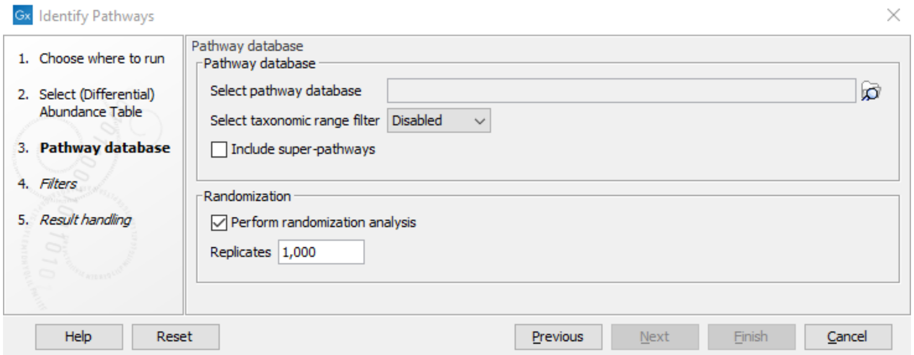
Toolbox | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Identify Pathways** ( )

Select a functional abundance table or a differential abundance table with EC terms as input and click "Next".

In the **Pathway database** section of the second step of the wizard (figure 11.20), select the required pathway database. A taxonomic range filter for the called pathways can be set to reduce the amount of false positive pathway calls in the case where the metagenomic reads are known to be of a certain type of origin. For example, if the (differential) abundance table has been produced from an OTU table based on ITS regions using Infer Functional Profile (beta), then the taxonomic range would have to be set to **Fungi**. Per default the filter is set to **Disabled** as is appropriate for many whole metagenome and metatranscriptome experiments. Finally, you can choose to include super-pathways in the analysis. This will have an influence on the minimum solution and the confidence scores as super-pathways are constructed of smaller pathways occurring in the pathway database. Since super pathways usually contain a lot of terms, it is more likely that a super-pathway is part of the minimum solution. Also, the super-pathway will tend to have a higher confidence at the cost of a lower confidence for the individual pathways it is composed of

In the **Randomization** section of the second wizard step (figure 11.20) it is possible to control the randomization experiment for setting the confidence scores. If **Perform randomization analysis** is selected, the order of pathways in the naive solution is shuffled and the pathways are called sequentially while removing their functional terms until no pathways or no functional terms are left. The number set for Replicates thereby controls how often this is executed and the confidence score becomes the fraction of randomizations in which a pathway is part of the solution. If the setting is deselected an estimate for this number will be given as the confidence of a pathway being present.

In the third wizard step (figure 11.21) it is possible to remove EC terms from the analysis based on the input table.



Identify Pathways

1. Choose where to run
2. Select (Differential) Abundance Table
3. **Pathway database**
4. *Filters*
5. *Result handling*

Pathway database

Pathway database

Select pathway database

Select taxonomic range filter: Disabled

□ Include super-pathways

Randomization

☑ Perform randomization analysis

Replicates: 1,000

Figure 11.20: *Select the pathway database, taxonomic range and set the randomization parameters.*

If the input table is an abundance table, the Abundance table filter section will be relevant. When selecting **Ignore terms with a low abundance value**, EC terms with abundance values below the value given in **Abundance threshold** will be ignored in the pathway calling procedure.

If the input table is a differential abundance table, several filters may be applied, one for each column for a statistical comparison in the differential abundance table. Note that some filters remove EC terms with values **lower** than the specified value in the corresponding field, i.e.

- Max Group Mean
- Absolute fold change
- Absolute log fold change

and other filters remove EC terms with with values **higher** than the specified value in the corresponding field

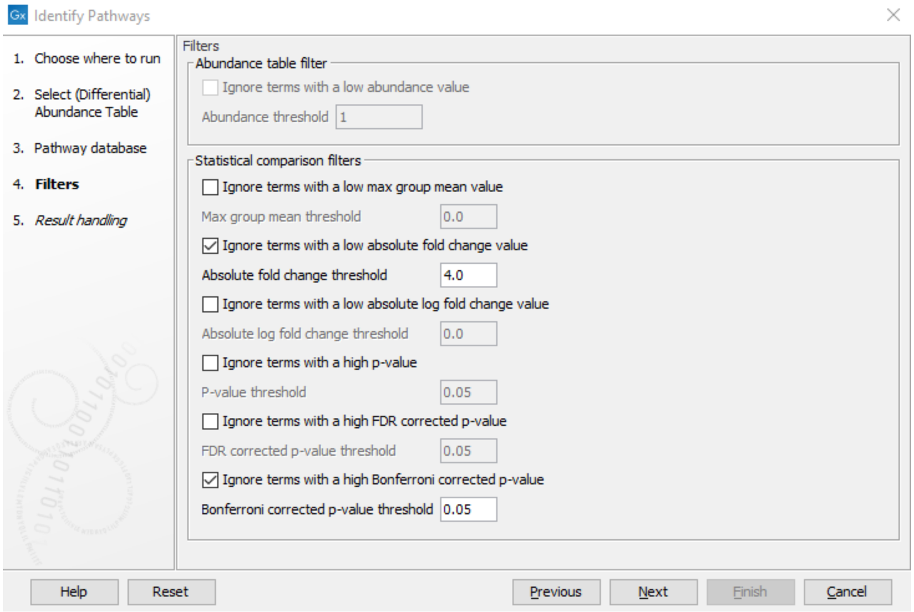
- P-value
- FDR corrected p-value
- Bonferroni corrected p-value

The filters may be combined freely to achieve the desired level of filtering. It is generally recommended to use a filter on the p-values, either FDR corrected or Bonferroni corrected to remove EC terms whose abundance level does not change between the groups. Based on the remaining terms after filtering, the naive, minimal and confidence based solutions will be calculated.

11.9.1 Called Pathways Result

The result of a Identify Pathways run is very similar to the pathway database, see section 16.3.1, in that it has three views:

- The Identified Pathways table ( )
- The Compound table ( )



Identify Pathways

1. Choose where to run
2. Select (Differential) Abundance Table
3. Pathway database
4. **Filters**
5. *Result handling*

Filters

Abundance table filter

☐ Ignore terms with a low abundance value

Abundance threshold: 1

Statistical comparison filters

☐ Ignore terms with a low max group mean value

Max group mean threshold: 0.0

☑ Ignore terms with a low absolute fold change value

Absolute fold change threshold: 4.0

☐ Ignore terms with a low absolute log fold change value

Absolute log fold change threshold: 0.0

☐ Ignore terms with a high p-value

P-value threshold: 0.05

☐ Ignore terms with a high FDR corrected p-value

FDR corrected p-value threshold: 0.05

☑ Ignore terms with a high Bonferroni corrected p-value

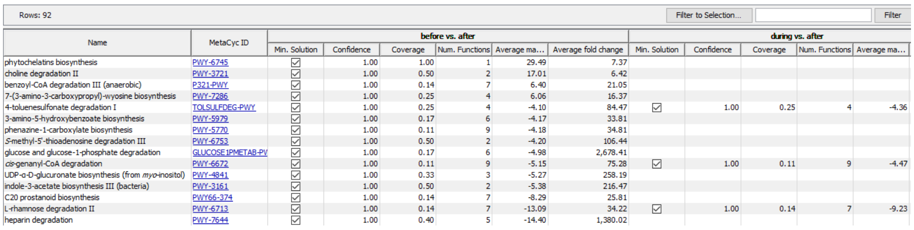
Bonferroni corrected p-value threshold: 0.05

Figure 11.21: *Filter the EC terms based on entries in the abundance table (Abundance table filter) or differential abundance table (Statistical Comparison filter) here shown for a differential abundance table.*

- The Enzyme table ( )

Note that the latter two are identical to the views in the pathway database, except that the pathways opened from these views are enriched with the data from (differential) abundance tables, see below.

The Identified Pathways Table The result of a Identify Pathways run presented as a table where each row corresponds to a pathway with a pathway name, pathway id and for each sample or comparison, depending on whether an abundance table or a differential abundance table has been used, a number of statistics on the pathway call (figure 11.22).



Rows: 92

Name	MetaCyc ID	before vs. after						during vs. after				
		Min. Solution	Confidence	Coverage	Num. Functions	Average ma...	Average fold change	Min. Solution	Confidence	Coverage	Num. Functions	Average ma...
phytochelatins biosynthesis	PWY-6745	☑	1.00	1.00	1	29.49	7.37					
choline degradation II	PWY-3721	☑	1.00	0.50	2	17.01	6.42					
benzoyl-CoA degradation III (anaerobic)	P321-PWY	☑	1.00	0.14	7	6.40	21.05					
7-(3-amino-3-carboxypropyl)-wyosine biosynthesis	PWY-7286	☑	1.00	0.25	4	6.06	16.37					
4-toluenesulfonate degradation I	TOLSULFDEG-PWY	☑	1.00	0.25	4	-4.10	84.47	☑	1.00	0.25	4	-4.36
3-amino-5-hydroxybenzoate biosynthesis	PWY-5979	☑	1.00	0.17	6	-4.17	33.81					
phenazine-1-carboxylate biosynthesis	PWY-5770	☑	1.00	0.11	9	-4.18	34.81					
S-methyl-5'-thioadenosine degradation III	PWY-6753	☑	1.00	0.50	2	-4.20	106.44					
glucose and glucose-1-phosphate degradation	GLUCOSE1PMETAB-PW	☑	1.00	0.17	6	-4.98	2,678.41					
cis-genanyl-CoA degradation	PWY-6672	☑	1.00	0.11	9	-5.15	75.28	☑	1.00	0.11	9	-4.47
UDP-α-D-glucuronate biosynthesis (from *myo*-inositol)	PWY-4841	☑	1.00	0.33	3	-5.27	258.19					
indole-3-acetate biosynthesis III (bacteria)	PWY-3161	☑	1.00	0.50	2	-5.38	216.47					
C20 prostanoid biosynthesis	PWY66-374	☑	1.00	0.14	7	-8.29	25.81					
L-rhamnose degradation II	PWY-6713	☑	1.00	0.14	7	-13.09	34.22	☑	1.00	0.14	7	-9.23
heparin degradation	PWY-7644	☑	1.00	0.40	5	-14.40	1,380.02					

Figure 11.22: *The result of the Identify Pathways tool is a table with pathways in the rows and some columns describing the pathway calls for each sample or comparison. Here the result is shown for a differential abundance table.*

There are two general columns to describe the pathways that have been called.

- **Name**: the name of the pathway.
- **MetaCyc ID**: the MetaCyc ID of the pathway, also a link to the corresponding MetaCyc page.

For each sample or comparison, there are four columns summarizing the result of the pathway calling procedure. Note that empty fields in this table mean that a pathway is not part of any solution for a given sample or comparison.

- **Min. Solution**: A check mark indicates whether the pathway is part of the minimal solution (see above), an unchecked checkbox means that the pathway is part of the naive solution (see above) and an empty field means that a pathway has not been identified at all.
- **Confidence**: A confidence score for the pathway to be called, given the EC terms after filtering. If **Perform randomization analysis** has been selected, the confidence score is calculated as the fraction of randomization experiments in which the pathway occurs. If the aforementioned option was not selected, a simple approximation for this number is given as confidence score. Typically, pathways which are part of the minimal solution also have a high confidence score, but not necessarily.
- **Coverage**: Reports the fraction of EC terms that have been identified (not filtered) of all EC terms that are present in that pathway.
- **Num. Functions**: Reports the number of EC terms present in a given pathway.

Depending on whether the input has been an abundance table or a differential abundance table, the result may contain some more columns giving average statistics for the EC terms from the (differential) abundance table for the whole pathway in which they occur. For an abundance table the column **Average abundance** gives the average abundance for all identified (not-filtered) EC terms that are present in the pathway. Similarly, for a differential abundance table the metrics are summarized by averaging over all identified (not-filtered) EC terms in a pathway, specifically the **Average max group mean** and **Average fold change** are reported.

Exporting content of the Identified Pathways table views The Identified Pathways Table can be exported to tabular format. To export the content of the Identified Pathways table view, run export with default parameters. To export the content of the Compound or Enzyme table view, take the following steps:

1. Open the The Identified Pathways Table.
2. Switch to the view you wish to export by clicking on the relevant icon below the view area.
3. While on the relevant view, launch the standard export functionality by clicking on the Export button in the Workbench toolbar or by selecting **Export** under the File menu.
4. Select the tabular format to export the data to.
5. Confirm the data element that has been pre-selected in the **Navigation Area**.
6. Configure the export parameters. Deselect **Export all columns**.
7. Select **Export table as currently shown**.
8. Select where the data should be exported to.
9. Click **Finish**.

11.9.2 The Identified Pathways View

When double clicking on a line in the table or selecting one or several lines and clicking on **Open Pathway View** in the bottom of the table, a simple visualization of the corresponding pathway(s) will be opened in a split view (figure 11.23). This visualization is similar to the pathway view of a pathway database, see section 16.3.2. This object has two views

- The Identified Pathways Graph View ()
- The Text Contents View ()

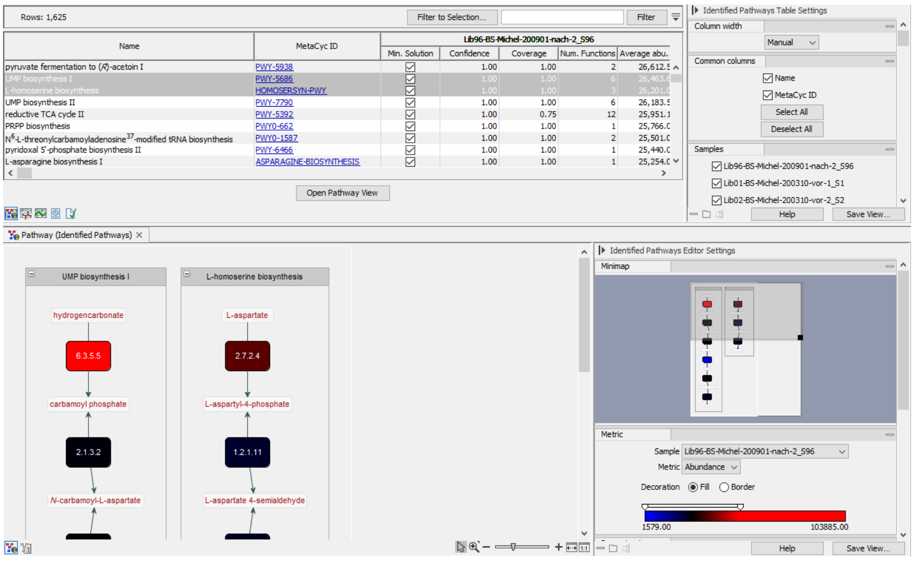
where the text contents view is the same as for the pathway view.

The Identified Pathways Graph View In this split view, the data from the original (differential) abundance table can be visualized on the EC terms. A minimap in the right upper corner of the side panel simplifies the navigation if many pathways have been opened at the same time. In the **Metric** section, a specific sample or comparison can be chosen for which the data will be visualized on the EC terms, whereas the Metric drop down menu provides a selection of different metrics associated with the EC terms. For an abundance table, the only option is Abundance whereas for a differential abundance table there are the options

- Max group mean
- Log_2 fold change
- Fold change
- P-value
- FDR p-value
- Bonferroni

corresponding to the headers in the differential abundance table.

In the lower right corner of the side panel there is a property viewer which displays information about selected EC terms and metabolites.



Rows: 1,625

Name	MetaCyc ID	Lib96-BS-Michel-200901-nach-2_S96				
		Min. Solution	Confidence	Coverage	Num. Functions	Average abu..
pyruvate fermentation to (*R*)-acetoin I	PWY-5938	☑	1.00	1.00	2	26,612.5
UMP biosynthesis I	PWY-5686	☑	1.00	1.00	6	26,463.6
L-homoserine biosynthesis	HOMOSERSYN-PWY	☑	1.00	1.00	3	26,201.0
UMP biosynthesis II	PWY-7790	☑	1.00	1.00	6	26,183.5
reductive TCA cycle II	PWY-5392	☑	1.00	0.75	12	25,951.1
PRPP biosynthesis	PWY0-662	☑	1.00	1.00	1	25,766.0
N^6-L-threonylcarbamoyladenosine^{37}-modified tRNA biosynthesis	PWY0-1587	☑	1.00	1.00	2	25,501.0
pyridoxal 5'-phosphate biosynthesis II	PWY-6466	☑	1.00	1.00	1	25,440.0
L-asparagine biosynthesis I	ASPARAGINE-BIOSYNTHESIS	☑	1.00	1.00	1	25,254.0

Identified Pathways Table Settings

Column width: Manual

Common columns: ☑ Name ☑ MetaCyc ID

Samples:
☑ Lib96-BS-Michel-200901-nach-2_S96
☑ Lib01-BS-Michel-200310-vor-1_S1
☑ Lib02-BS-Michel-200310-vor-2_S2

Identified Pathways Editor Settings

Metric
Sample: Lib96-BS-Michel-200901-nach-2_S96
Metric: Abundance
Decoration: ● Fill ○ Border

Figure 11.23: *A pathway call result from an abundance table, where the abundances from the input table correspond to the hue of the functional entities in the pathway. The hues or width of the functional terms in the pathway visualization can be set to any metric from the input (differential) abundance table for a given sample or comparison.*

Chapter 12

Drug Resistance Analysis

Antimicrobial resistance (AMR) is an emerging threat to human and animal health worldwide. It is therefore of great importance to identify antimicrobial resistance genes from isolate or metagenomic sequencing data both for monitoring purposes and to gain actionable insight. To detect resistances from sequencing reads directly, the tools **Find Resistance with ShortBRED** and **Find Resistance with PointFinder** can be used for probing the presence of resistance gene families or resistance conferring mutations, respectively. In order to detect AMR genes from assembled sequencing reads, the tool **Find Resistance with Nucleotide Database** is available.

12.1 Find Resistance with PointFinder

Find Resistance with PointFinder identifies known antimicrobial resistance conferring mutations from reads. In contrast to the Find Resistance with Nucleotide Database tool that quantifies the occurrence of entire resistance conferring genes, the aim is to detect the presence of resistance conferring mutations in antimicrobial targets in both susceptible and resistant strains.

The presence of antimicrobial resistance conferring variants can be inferred by mapping reads to a Pointfinder database containing both wild type and known resistant mutants of antimicrobial target genes.

PointFinder databases can be downloaded using **Download Resistance Database** (section 17.1).

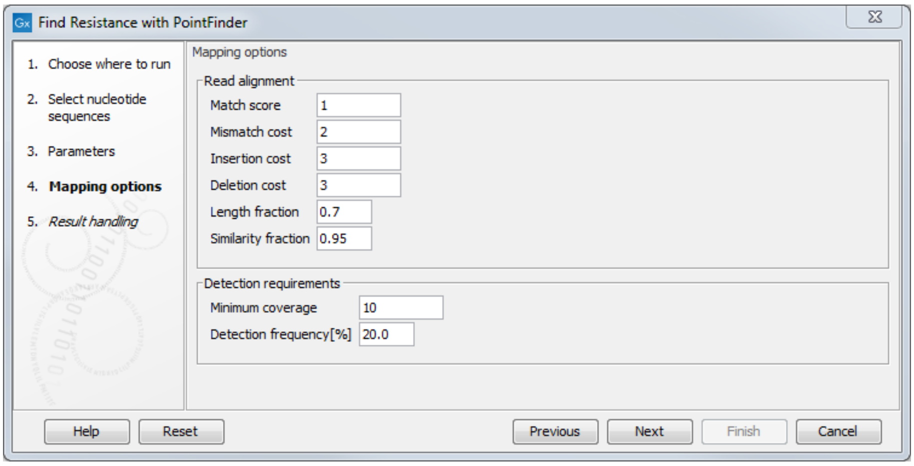
To run Find Resistance with PointFinder, go to:

> **Toolbox** | **Microbial Genomics Module** () | **Drug Resistance Analysis** () | **Find Resistance with PointFinder** ()

The tool accepts a nucleotide sequence or sequence list as input, followed by the selection of the PointFinder Database for the relevant pathogen.

For the read mapping step, several parameters are available (figure 12.1):

- **Match score**: score for a match
- **Mismatch cost**: cost for a mismatch
- **Insertion cost**: cost for an insertion
- **Deletion cost**: cost for accepting a deletion



Find Resistance with PointFinder

1. Choose where to run
2. Select nucleotide sequences
3. Parameters
4. **Mapping options**
5. *Result handling*

Mapping options

Read alignment

Match score: 1
Mismatch cost: 2
Insertion cost: 3
Deletion cost: 3
Length fraction: 0.7
Similarity fraction: 0.95

Detection requirements

Minimum coverage: 10
Detection frequency[%]: 20.0

Figure 12.1: *Options.*

- **Length fraction**: minimum length of the mapped portion of the read
- **Similarity fraction**: minimal fraction of matches in the mapped region

The tool first maps the reads to the specified database sequences. Next, the tool analyses the read mappings. For each reference sequence containing the variant, the tool verifies if the reads mapping to that reference or related references (e.g. references with an additional mutation) contain that variant. The result table will only list those entries from the database that have a minimum coverage and exceed the minimum frequency threshold. The tool also adds information:

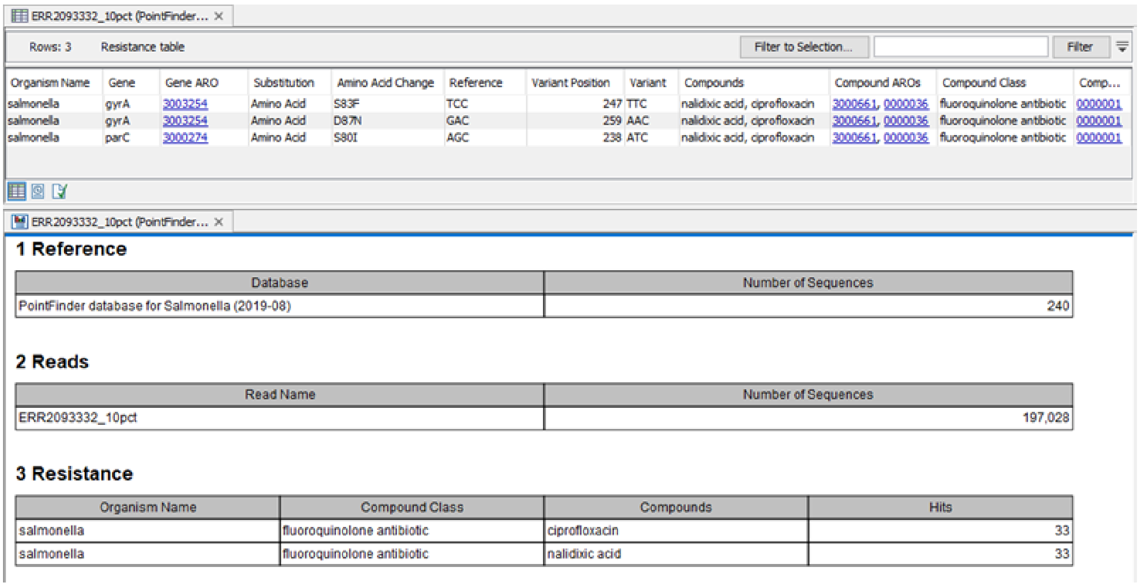
- **Minimum coverage**: the number of reads covering the variant region. The coverage must be complete, e.g. all 3 nucleotides that constitute an amino acid change have to be covered.
- **Detection frequency**: The minimum allele frequency that is required to annotate a variant as being present in the sample.

The tool outputs a table containing information about the variants detected in the reads. The columns available in that table can be seen in figure 12.2 (which also shows an example of report output by the tool):

By enabling the "Output annotated reads" option, the user can obtain a copy of the subset of reads that map to target variants. The reads will be annotated with the following annotations:

- Target: The name of the target reference (from the PointFinder database) where the read mapped to, e.g. "salmonella_gyrA_AAS_S_83_Y_TCC_247_TAC".
- Compound Class: The compound class to which the variant gives resistance, e.g. fluoroquinolone antibiotic.
- Compounds: The compound(s) class to which the variant gives resistance, e.g. colistin.

The resulting report contains information about the reads and the database that was used.


ERR2093332_10pct (PointFinder...
Filter to Selection...
Filter
ERR2093332_10pct (PointFinder...

Rows: 3 Resistance table

Organism Name	Gene	Gene ARO	Substitution	Amino Acid Change	Reference	Variant Position	Variant	Compounds	Compound AROs	Compound Class	Comp...
salmonella	gyrA	3003254	Amino Acid	S83F	TCC	247	TTC	nalidixic acid, ciprofloxacin	3000661, 0000036	fluoroquinolone antibiotic	0000001
salmonella	gyrA	3003254	Amino Acid	D87N	GAC	259	AAC	nalidixic acid, ciprofloxacin	3000661, 0000036	fluoroquinolone antibiotic	0000001
salmonella	parC	3000274	Amino Acid	S80I	AGC	238	ATC	nalidixic acid, ciprofloxacin	3000661, 0000036	fluoroquinolone antibiotic	0000001

1 Reference

Database	Number of Sequences
PointFinder database for Salmonella (2019-08)	240

2 Reads

Read Name	Number of Sequences
ERR2093332_10pct	197,028

3 Resistance

Organism Name	Compound Class	Compounds	Hits
salmonella	fluoroquinolone antibiotic	ciprofloxacin	33
salmonella	fluoroquinolone antibiotic	nalidixic acid	33

Figure 12.2: *Table generated by Find Resistance with PointFinder, shown here together with the corresponding report.*

12.2 Find Resistance with Nucleotide Database

Identification of antimicrobial resistance genes is important for understanding the underlying mechanisms and the epidemiology of antimicrobial resistance. The Find Resistance with Nucleotide Database tool may be used for resistance typing of pre-assembled, complete or partial genomes simple contig sequences assembled using the de novo assembly algorithm of CLC Genomics Workbench (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=De_novo_assembly.html).

Alternatively, use the **Type a Known Species** or **Type among Multiple Species** template workflows as described in chapter 2.

Find Resistance with Nucleotide Database is inspired by Zankari et al., 2017 and uses BLAST for identification of acquired antimicrobial resistance genes within whole-genome sequencing (WGS) data. The tool detects resistance conferring genes, whether they break down (e.g. beta-lactamases) or expel (e.g. efflux-pumps) antimicrobial compounds.

Nucleotide resistance databases for use with the tool can be downloaded using **Download Resistance Database** (section 17.1).

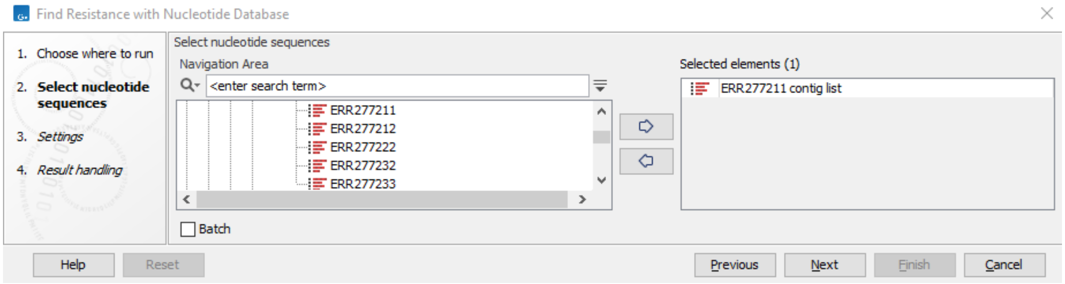
To perform resistance typing, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Drug Resistance Analysis** ( ) | **Find Resistance with Nucleotide Database** ( )

Select the input genome or contigs (figure 12.3).

You can then specify the settings for the tool (figure 12.4).

- **Database** Select a nucleotide resistance database.
- **Minimum Identity %** is the threshold for the minimum percentage of nucleotides that are identical between the best matching resistance gene in the database and the corresponding sequence in the genome.



Find Resistance with Nucleotide Database

1. Choose where to run
2. **Select nucleotide sequences**
3. *Settings*
4. *Result handling*

Select nucleotide sequences

Navigation Area

<enter search term>

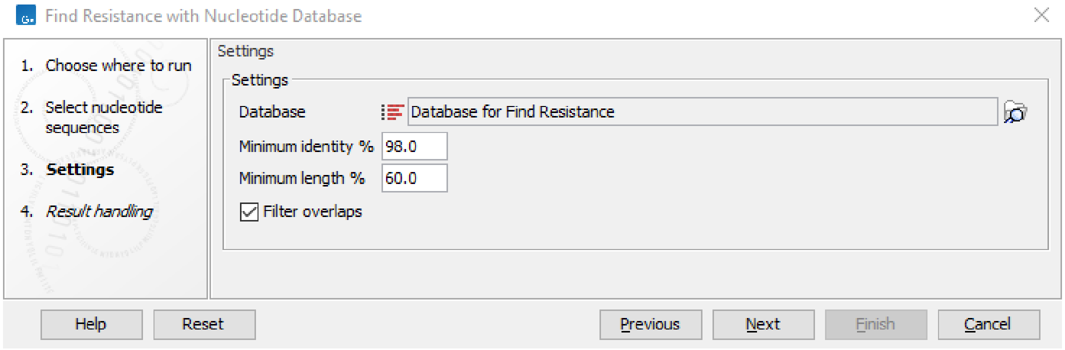
ERR277211
ERR277212
ERR277222
ERR277232
ERR277233

Selected elements (1)

ERR277211 contig list

□ Batch

Figure 12.3: *Pre-assembled and complete- or partial genomes simple contig sequences may be used as input for resistance typing.*



Find Resistance with Nucleotide Database

1. Choose where to run
2. Select nucleotide sequences
3. **Settings**
4. *Result handling*

Settings

Settings

Database: Database for Find Resistance

Minimum identity %: 98.0

Minimum length %: 60.0

☑ Filter overlaps

Figure 12.4: *Select database and settings for resistance typing.*

- **Minimum Length %** reflect the percentage of the total resistance gene length that a sequence must overlap a resistance gene to count as a hit for that gene. Here represented as a percentage of the total resistance gene length.
- **Filter overlaps**: will perform extra filtering of results per contig, where one hit is contained by the other with a preference for the hit with the higher number of aligned nucleotides (length * identity).

The output of the Find Resistance with Nucleotide Database tool is a table listing all the possible resistance genes and predicted phenotypes found in the input genome or contigs, as well as additional information such as degrees of similarity between the gene found in the genome and the reference (% identity and query /HSB values) and the location where the gene was found (contig name, and position in the contig). Depending on the type of database used, additional columns with link to resources may also be present in the table. To add the obtained resistance types to your Result Metadata Table, see section 19.2.3.

12.3 Find Resistance with ShortBRED

This tool allows you to detect and quantify the presence of antibiotic resistance (AR) marker genes by running DIAMOND with a database of peptide marker sequences which represent the genes of interest. The Find Resistance with ShortBRED tool works similarly to the quantify step of ShortBRED, a public bioinformatics pipeline and resource. To learn more about the ShortBRED-Quantify tool [Kaminski et al., 2015], see `https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004557`.

Find Resistance with ShortBRED quantifies the presence of Antibiotic Resistance (AR) marker genes in a sample of NGS short reads. It is possible to output a sequence list containing all the input reads which contained a marker (each read in the output is annotated with metadata describing the properties of the marker detected in the read).

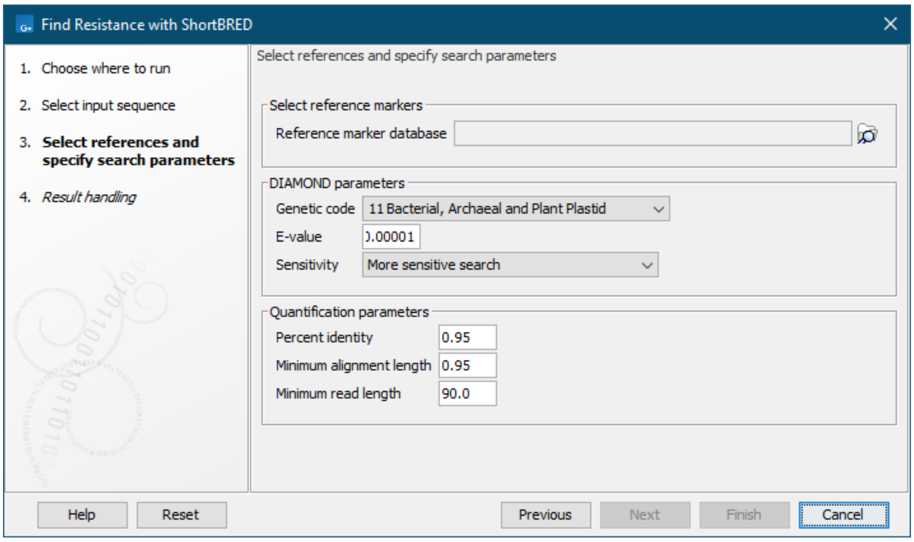
Antibiotic Resistance marker databases for use with the tool can be downloaded using **Download Resistance Database** (section 17.1).

To start the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Drug Resistance Analysis** ( ) | **Find. Resistance with ShortBRED** ( )

The tool accepts a nucleotide sequence or sequence list as input.

In the next dialog (figure 12.5), several parameters are available:



Find Resistance with ShortBRED

1. Choose where to run
2. Select input sequence
3. **Select references and specify search parameters**
4. *Result handling*

Select references and specify search parameters

Select reference markers

Reference marker database

DIAMOND parameters

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

E-value:).00001

Sensitivity: More sensitive search

Quantification parameters

Percent identity: 0.95

Minimum alignment length: 0.95

Minimum read length: 90.0

Figure 12.5: *References and search parameters.*

- **Reference marker database**: select the antimicrobial resistance marker database.
- **Genetic code**: select the genetic code to use when DIAMOND translates the nucleotide sequences to proteins.
- **E-value**: specify an expectation value to use as threshold for qualifying hits with DIAMOND.
- **Sensitivity**: select DIAMOND sensitivity setting.
- **Percent identity**: defines a minimum threshold for the percent identity of an alignment. This value is used by Find Resistance with ShortBRED to determine whether a hit found by DIAMOND is sufficiently good to be validated as a true hit (equivalent to the parameter "-id" of the ShortBRED-Quantify tool).
- **Minimum alignment length**: the minimum length of the DIAMOND alignment. This value is used by Find Resistance with ShortBRED to determine whether a hit found by DIAMOND is sufficiently good to be validated as a true hit (equivalent to the parameter "-pctlength" of the ShortBRED-Quantify tool - see citation at the beginning of this section).
- **Minimum read length**: the minimum read length. This value is used by Find Resistance with ShortBRED to determine whether a read is long enough to be processed by Find Resistance with ShortBRED (equivalent to the parameter "-minreadBP" of the ShortBRED-Quantify tool).

The Find Resistance with ShortBRED tool will output a resistance abundance table, a result summary table, an optional report, and an optional sequence list.

The result summary table provides an overview of all the resistance phenotypes in the applied database and reports the number of identified resistance genes and markers, and number of reads assigned to each phenotype.

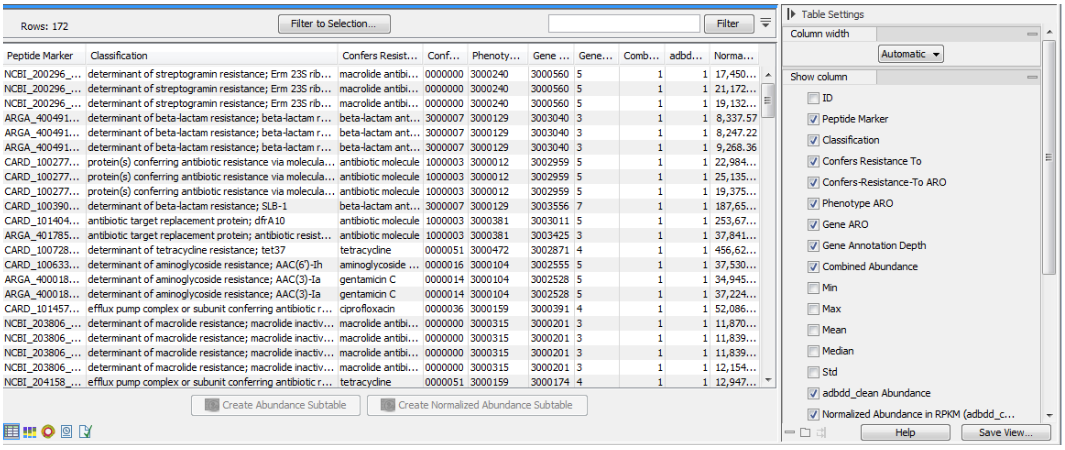
The optional report output contains general information about the input sample, the marker database used and a short result summary.

The optional sequence list output contains all reads from the input sample which were found to contain one of the AR marker sequences. Each read in the sequence list is annotated with metadata describing the properties of the marker detected in the read.

12.3.1 Resistance abundance table

The Resistance abundance table summarizes the abundance of each marker, i.e., it reports the number of times a given marker is found in the input reads. The abundance is also reported in units of RPKM, referred to as the normalized abundance, which are calculated in the same way as is done by ShortBRED-Quantify. It is possible to aggregate the abundance by gene name and resistance phenotype to get the abundance at each of these levels. The table can be visualized using the Stacked bar charts and Stacked area charts function, as well as the Sunburst charts.

- **Table view** (▦) (figure 12.6)



Rows: 172

Peptide Marker	Classification	Confers Resist...	Conf...	Phenoty...	Gene ...	Gene...	Comb...	adbd...	Norma...
NCBI_200296_...	determinant of streptogramin resistance; Erm 23S rib...	macrolide antibi...	0000000	3000240	3000560	5	1	1	17,450...
NCBI_200296_...	determinant of streptogramin resistance; Erm 23S rib...	macrolide antibi...	0000000	3000240	3000560	5	1	1	21,172...
NCBI_200296_...	determinant of streptogramin resistance; Erm 23S rib...	macrolide antibi...	0000000	3000240	3000560	5	1	1	19,132...
ARGA_400491...	determinant of beta-lactam resistance; beta-lactam r...	beta-lactam ant...	3000007	3000129	3003040	3	1	1	8,337.57
ARGA_400491...	determinant of beta-lactam resistance; beta-lactam r...	beta-lactam ant...	3000007	3000129	3003040	3	1	1	8,247.22
ARGA_400491...	determinant of beta-lactam resistance; beta-lactam r...	beta-lactam ant...	3000007	3000129	3003040	3	1	1	9,268.36
CARD_100277...	protein(s) conferring antibiotic resistance via molecula...	antibiotic molecule	1000003	3000012	3002959	5	1	1	22,984...
CARD_100277...	protein(s) conferring antibiotic resistance via molecula...	antibiotic molecule	1000003	3000012	3002959	5	1	1	25,135...
CARD_100277...	protein(s) conferring antibiotic resistance via molecula...	antibiotic molecule	1000003	3000012	3002959	5	1	1	19,375...
CARD_100390...	determinant of beta-lactam resistance; SLB-1	beta-lactam ant...	3000007	3000129	3003556	7	1	1	187,65...
CARD_101404...	antibiotic target replacement protein; dfrA10	antibiotic molecule	1000003	3000381	3003011	5	1	1	253,67...
ARGA_401785...	antibiotic target replacement protein; antibiotic resist...	antibiotic molecule	1000003	3000381	3003425	3	1	1	37,841...
CARD_100728...	determinant of tetracycline resistance; tet37	tetracycline	0000051	3000472	3002871	4	1	1	456,62...
CARD_100633...	determinant of aminoglycoside resistance; AAC(6')-Ih	aminoglycoside ...	0000016	3000104	3002555	5	1	1	37,530...
ARGA_400018...	determinant of aminoglycoside resistance; AAC(3)-Ia	gentamicin C	0000014	3000104	3002528	5	1	1	34,945...
ARGA_400018...	determinant of aminoglycoside resistance; AAC(3)-Ia	gentamicin C	0000014	3000104	3002528	5	1	1	37,224...
CARD_101457...	efflux pump complex or subunit conferring antibiotic r...	ciprofloxacin	0000036	3000159	3000391	4	1	1	52,086...
NCBI_203806_...	determinant of macrolide resistance; macrolide inactiv...	macrolide antibi...	0000000	3000315	3000201	3	1	1	11,870...
NCBI_203806_...	determinant of macrolide resistance; macrolide inactiv...	macrolide antibi...	0000000	3000315	3000201	3	1	1	11,839...
NCBI_203806_...	determinant of macrolide resistance; macrolide inactiv...	macrolide antibi...	0000000	3000315	3000201	3	1	1	11,839...
NCBI_203806_...	determinant of macrolide resistance; macrolide inactiv...	macrolide antibi...	0000000	3000315	3000201	3	1	1	12,154...
NCBI_204158_...	efflux pump complex or subunit conferring antibiotic r...	tetracycline	0000051	3000159	3000174	4	1	1	12,947...

Create Abundance Subtable
Create Normalized Abundance Subtable

Table Settings

Column width: Automatic

Show column

- □ ID
- ☑ Peptide Marker
- ☑ Classification
- ☑ Confers Resistance To
- ☑ Confers-Resistance-To ARO
- ☑ Phenotype ARO
- ☑ Gene ARO
- ☑ Gene Annotation Depth
- ☑ Combined Abundance
- □ Min
- □ Max
- □ Mean
- □ Median
- □ Std
- ☑ adbdd_clean Abundance
- ☑ Normalized Abundance in RPKM (adbdd_c...

Figure 12.6: *Resistance abundance table.*

The table displays the following columns (note that the columns can vary depending on the marker database used):

- **ID**: internal ID which the abundance tables use for ordering the samples. IDs are unique, so that when merging abundance tables peptide markers with the same ID will be combined.
- **Peptide Marker**: the name of the Peptide Marker as it is given in the AR marker database.
- **Classification**: the Classification of the peptide marker contains the resistance phenotype and the gene name separated by a semi-colon.
- **Confers Resistance To**: the antibiotic class which this marker confers resistance to.

  - **Confers-Resistance-To ARO**: the ARO (Antibiotic Resistance Ontology) ID number of the "Confers-Resistance-To" property.
  - **Phenotype ARO**: the ARO ID number associated with this particular resistance phenotype.
  - **Gene ARO**: the ARO ID number associated with this particular gene.
  - **Gene Annotation Depth**: the annotation depth of the gene name in the Antibiotic Resistance Ontology. The higher the number the more specific is the annotation.
  - **Combined Abundance**: reports the number of times a given marker is found in the input reads across all samples.
  - **Min**, **Max**, **Mean**, **Median** and **Std**: respectively minimum, maximum, mean, median and standard deviation of the number of reads for the taxa across all samples.
  - **Name of the sample Abundance** (for example SRR2754560 in the table above): number of reads containing this peptide marker for each sample.
  - **Normalized Abundance in RPKM** (name of the sample): Normalized Abundance is reported in units of RPKM (Reads Per Kilobase per Million reads) which are calculated in the same way as is done by ShortBRED-Quantify.

  In the right side panel, under the tab Data, you can switch between raw and relative abundances. You can also combine absolute counts and relative abundances by Phenotype and Gene name by selecting the appropriate level in the **Aggregate feature** drop-down menu. Incomplete fatures at a given level of Aggregation can be hidden using the "Hide incomplete features" check box.

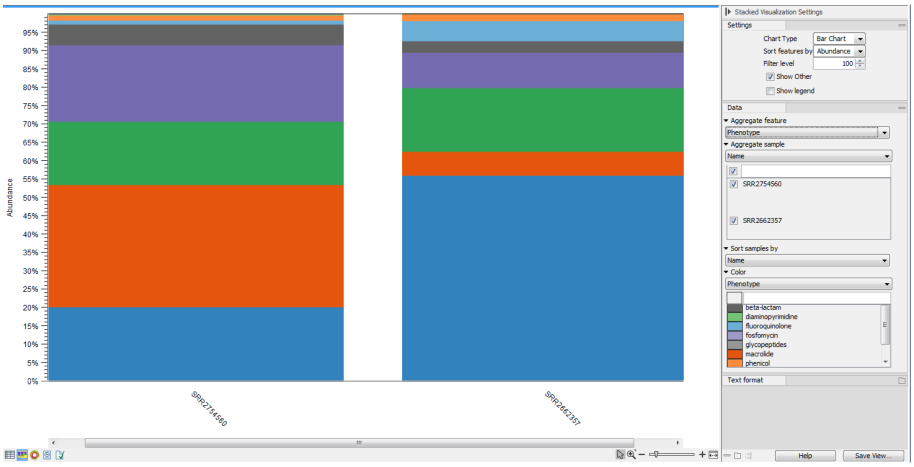
  Finally, if you have previously annotated your table with Metadata (see section 6.8), you can **Aggregate sample** by the groups previously defined in your metadata table. This is useful when for example analyzing replicates from the same sample origin.

  Above and under the table, the following actions are available:

  - **Filter to Selection...** to have the table only displaying pre-selected rows in the table.
  - **Create Abundance Subtable** will create a table containing only the selected rows.
  - **Create Normalized Abundance Subtable** will create a table with all rows normalized on the values of a single selected row. The row used for normalization will disappear from the new abundance table. The normalization scales the abundance table linearly, where the scaling factor is calculated by determining the average abundance across all samples and for each sample scale it to the average for the reference. If you have zero values in some samples for the control, you will need to generate a new abundance table where these samples are not present. If the abundance table is obtained from merging single-sample abundance tables, then the merge should be redone excluding the samples with zero control read counts.

- **Stacked Bar Chart and Stacked Area Chart** (▦)

  Choose which chart you want to see using the drop down menu in the upper right corner of the side panel. In the Stacked Bar (figure 12.7) and Stacked Area Charts (figure 12.8), the metadata can be used to aggregate groups of columns (samples) by selecting the relevant metadata category in the right hand side panel. Also, the data can be aggregated at any classification level selected. The relevant data points will automatically be summed accordingly.



Stacked Visualization Settings

Settings

Chart Type: Bar Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

□ Show legend

Data

Aggregate feature: Phenotype

Aggregate sample: Name

☑ SRR2754560

☑ SRR2662357

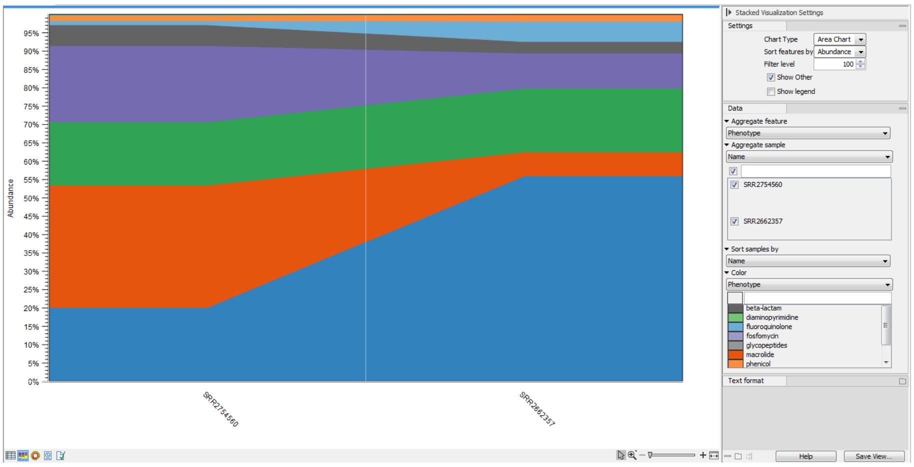
Sort samples by: Name

Color: Phenotype

beta-lactam, diaminopyrimidine, fluoroquinolone, fosfomycin, glycopeptides, macrolide, phenicol

Text format

Figure 12.7: *Stacked bar chart.*



Stacked Visualization Settings

Settings

Chart Type: Area Chart

Sort features by: Abundance

Filter level: 100

☑ Show Other

□ Show legend

Data

Aggregate feature: Phenotype

Aggregate sample: Name

☑ SRR2754560

☑ SRR2662357

Sort samples by: Name

Color: Phenotype

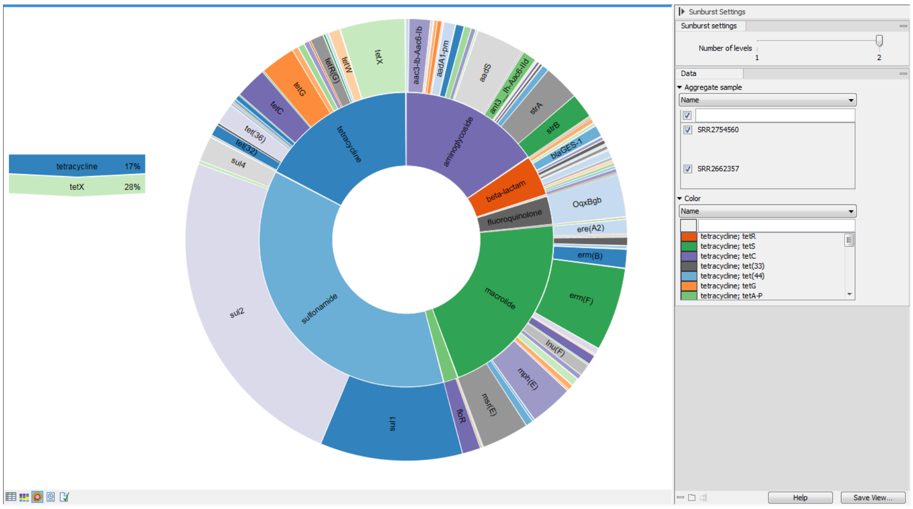
beta-lactam, diaminopyrimidine, fluoroquinolone, fosfomycin, glycopeptides, macrolide, phenicol

Text format

Figure 12.8: *Stacked area chart.*

Holding the pointer over a colored area in any of the plots will result in the display of the corresponding classification label and counts. **Filter level** allows to modify the number of features to be shown in the plot. For example, setting the value to 10 means that the 10 most abundant features of each sample will be shown in all columns. The remaining features are grouped into "Other", and will be shown if the option is selected in the right hand side panel. One can select which classification level to color, and change the default colors manually. Colors can be specified at the same classification level as the one used to aggregate the data or at a lower level. When lower classification levels are chosen in the data aggregation field, the color will be inherited in alternating shadings. It is also possible to sort samples by metadata attributes, and to show groups of samples without collapsing their stacks, as well as change the label of each stack or group of stacks. Features can be sorted by "abundance" or "name" using the drop down menu in the right hand side panel. Using the bottom right-most button (**Save/restore settings** ( )), the settings can be saved and applied in other plots, allowing visual comparisons across analyses.

- **Zoomable Sunbursts** ( ) The Zoomable Sunburst viewer lets the user select how many classification level counts to display, and which level to color. Lower levels will inherit the color in alternating shadings. Classification and relative abundances are displayed in a legend to the left of the plot when hovering over the sunburst viewer with the mouse. The metadata can be used to select which sample or group of samples to show in the sunburst (figure 12.9).



Sunburst Settings

Sunburst settings

Number of levels 1 2

Data

Aggregate sample

Name

☑

☑ SRR2754560

☑ SRR2662357

Color

Name

tetracycline; tetR
tetracycline; tetS
tetracycline; tetC
tetracycline; tet(33)
tetracycline; tet(44)
tetracycline; tetG
tetracycline; tetA-P

Figure 12.9: *Sunburst view.*

Clicking on a lower level field will render that field the center of the plot and display lower level counts in a radial view. Clicking on the center field will render the level above the current view the center of the view.

Part VI

Databases

Chapter 13

Databases for MLST Schemes

13.1 Create MLST Scheme

The **Create MLST Scheme** tool can be used to create a scheme from scratch.

To run the **Create MLST Scheme** tool choose:

> **Toolbox** | **Microbial Genomics Module** **( )** | **Databases** ( ) | **MLST Typing** ( ) | **Create MLST Scheme** ( )

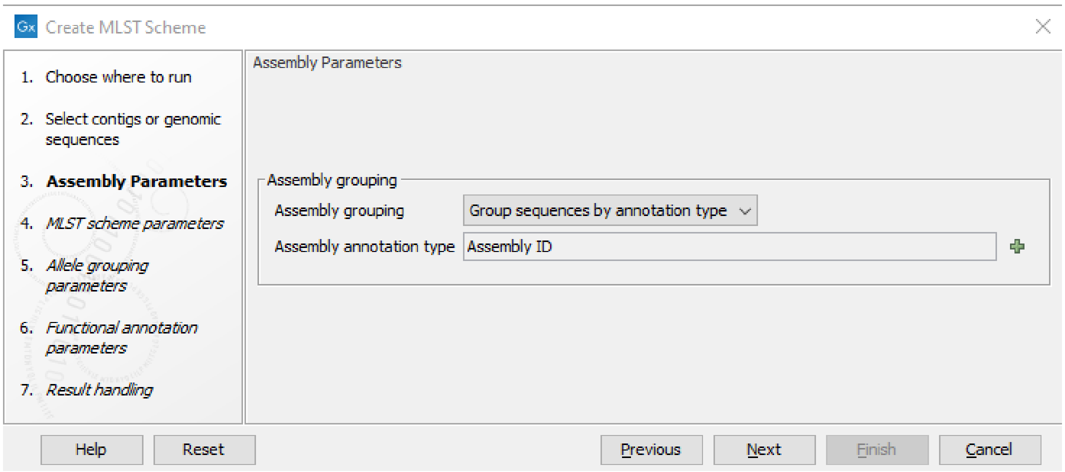
As input, the tool requires a set of complete isolate genomes in the form of one or more sequence lists or sequences. At least one of these genomes must be annotated with coding region (CDS) annotations. If these are not available, the Find Prokaryotic Genes tool (see section 11.1) or Annotate with DIAMOND (see section 11.3) can be used to predict and annotate the coding regions.

In the first wizard step shown in figure 13.1 the grouping of sequences into genomic units can be controlled. This is necessary when working with genomes that span several chromosomes or several contigs for the tool to consider these as one unit. The grouping can be controlled by the **Assembly grouping** field:

- Each sequence is one assembly: Each individual sequence is considered a complete assembly of a genome.
- Each input element is one assembly: Each input element, i.e. input sequence or input sequence list, is considered a complete assembly of a genome.
- Group sequences by annotation type: Use annotations to group the assemblies and specify the annotation field with **Assembly annotation type**. Some tools, such as the **Download Custom Microbial Reference Database**, will automatically assign an Assembly ID that can be used for grouping. For a manual assignment of Assembly ID annotations, please see section 21.

After specifying the input, the second step is to set up the basic MLST Scheme creation parameters (figure 13.2).

The **Create MLST Scheme** tool works by extracting all annotated coding sequences (CDS) and clustering them into similar gene classes (loci). It is possible to specify whether we are interested


Create MLST Scheme
1. Choose where to run
2. Select contigs or genomic sequences
3. Assembly Parameters
4. MLST scheme parameters
5. Allele grouping parameters
6. Functional annotation parameters
7. Result handling
Help
Reset
Previous
Next
Finish
Cancel

Assembly Parameters

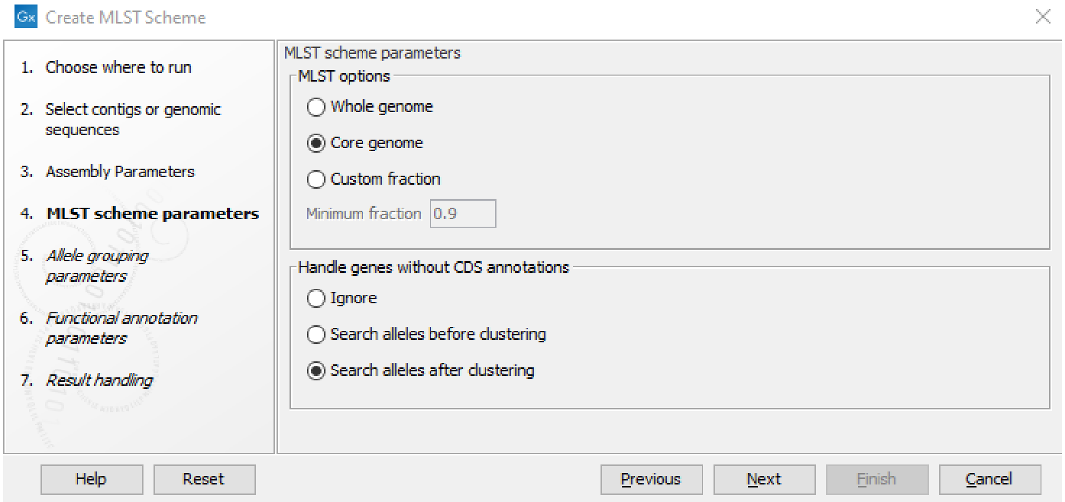
Assembly grouping

Assembly grouping: Group sequences by annotation type

Assembly annotation type: Assembly ID

Figure 13.1: *Grouping the input into assemblies.*

in the genes that are present in some genomes (**Whole genome** - must be present in at least 10% of all genomes), most genomes (**Core genome** - must be present in at least 90% of the genomes), or a user-specified **Minimum fraction**.


Create MLST Scheme
1. Choose where to run
2. Select contigs or genomic sequences
3. Assembly Parameters
4. MLST scheme parameters
5. Allele grouping parameters
6. Functional annotation parameters
7. Result handling
Help
Reset
Previous
Next
Finish
Cancel

MLST scheme parameters

MLST options

○ Whole genome

● Core genome

○ Custom fraction

Minimum fraction 0.9

Handle genes without CDS annotations

○ Ignore

○ Search alleles before clustering

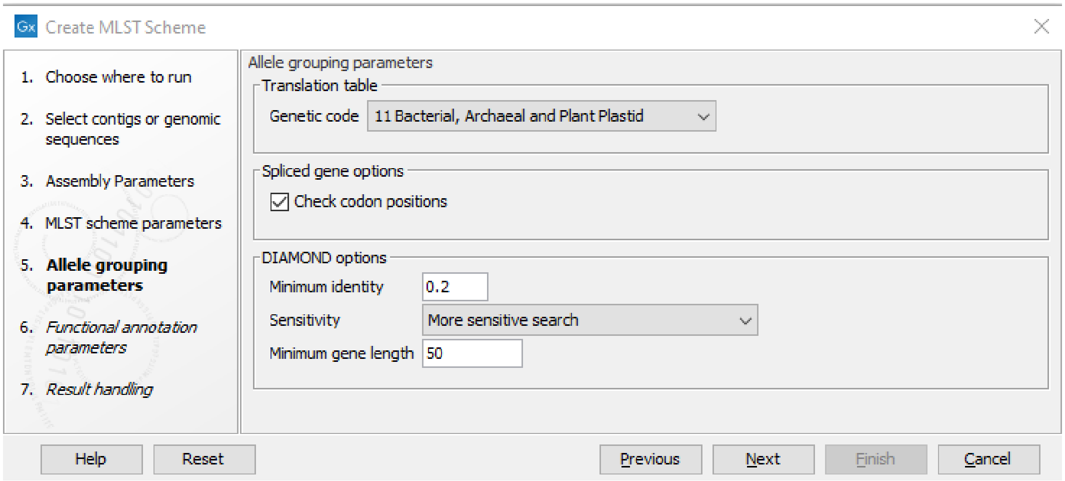
● Search alleles after clustering

Figure 13.2: *Basic options for creating a MLST scheme.*

The best results are obtained by supplying genomes with proper CDS annotations. The **Handle genes without annotations** option controls how genomes without CDS annotations and how existing CDS may be overridden if a longer CDS from another genome exactly matches the genomic sequence.

- **Ignore**: Only use the existing CDS annotations as a basis for the MLST scheme construction.
- **Search alleles before clustering**: All of the input genomes are blasted (using DIAMOND) against the set of annotated genes, and any new genes will be added as alleles. This is a very slow, but thorough check.
- **Search alleles after clustering**: After clustering the genes, all of the input genomes are blasted (using DIAMOND), but only against the longest protein in each cluster.

The **Allele grouping parameters** step (figure 13.3) specifies how the different genes (CDS annotations) are compared to each other. DIAMOND is used for this clustering. The following can be specified:



1. Choose where to run
2. Select contigs or genomic sequences
3. Assembly Parameters
4. MLST scheme parameters
5. **Allele grouping parameters**
6. *Functional annotation parameters*
7. *Result handling*

Allele grouping parameters

Translation table

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Spliced gene options

☑ Check codon positions

DIAMOND options

Minimum identity: 0.2

Sensitivity: More sensitive search

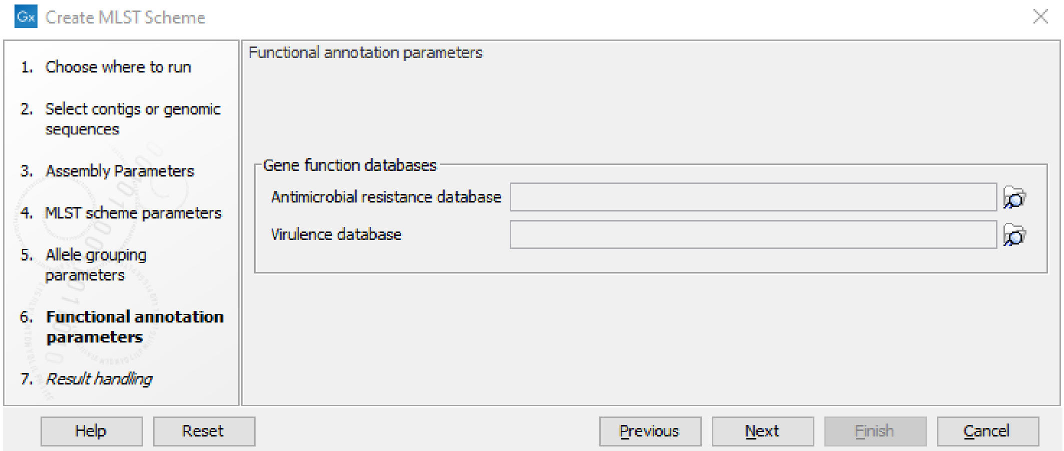
Minimum gene length: 50

Figure 13.3: *The allele grouping (clustering) options.*

- **Genetic code**: specifies the genomic code to use for the input samples if **Check codon positions** is enabled.
- **Check codon positions**: If this is enabled, coding sequences not starting with a start codon, not ending with a stop codon or containing internal stop codons will be discarded. This can be disabled, for example to allow the construction of MLST schemes with spliced genes where each exon is considered an allele.
- **Minimum identity**: determines the minimum sequence identity before grouping protein sequences.
- It is also possible to specify the sensitivity of the search (**Standard search**, **Sensitive search**, and **More sensitive search**) - increasing the sensitivity makes the search more thorough, but also much slower. The default for this parameter is **Sensitive search**.
- **Minimum gene length**: is used to remove short genes from the resulting MLST scheme.

Note that after clustering, length outliers of a given cluster are removed by applying Tukey's fences with an interquartile range of 1.5, yet allowing for 5% length variation around the median. For example, for an allele cluster (locus) with allele lengths 51, 51, 51, 51, 53, the latter allele will not be removed although it falls outside the 1.5 IQR (both the first and third quartile are 51) since it is still within 5% of the median, for 48, 51, 51, 54, 63, only the former four will be included.

It is possible to decorate the alleles with information about virulence or resistance. The information can be extracted from either a ShortBRED Marker database or a Nucleotide database. These databases can be accessed using the Download Resistance Database tool (section 17.1) and can be provided as input to the **Create MLST Scheme** tool at this step (figure 13.4).


Create MLST Scheme
1. Choose where to run
2. Select contigs or genomic sequences
3. Assembly Parameters
4. MLST scheme parameters
5. Allele grouping parameters
6. Functional annotation parameters
7. Result handling
Help
Reset
Previous
Next
Finish
Cancel

Functional annotation parameters

Gene function databases

Antimicrobial resistance database

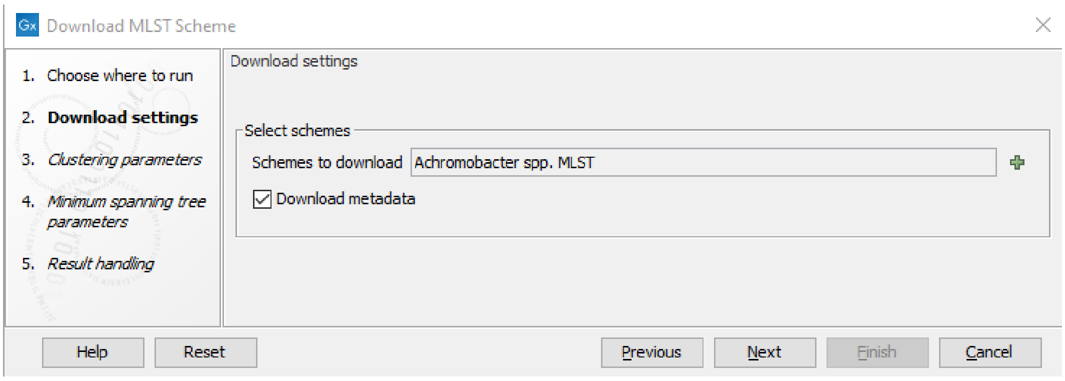
Virulence database

Figure 13.4: *The functional annotation parameters.*

13.2 Download MLST Scheme

To run the **Download MLST Scheme** tool choose:

Toolbox | **Microbial Genomics Module** ( ) | **Databases** ( ) | **MLST Typing** ( ) | **Download MLST Scheme** ( )



Download settings

Select schemes

Schemes to download Achromobacter spp. MLST

☑ Download metadata

Figure 13.5: *The Download MLST Scheme settings.*

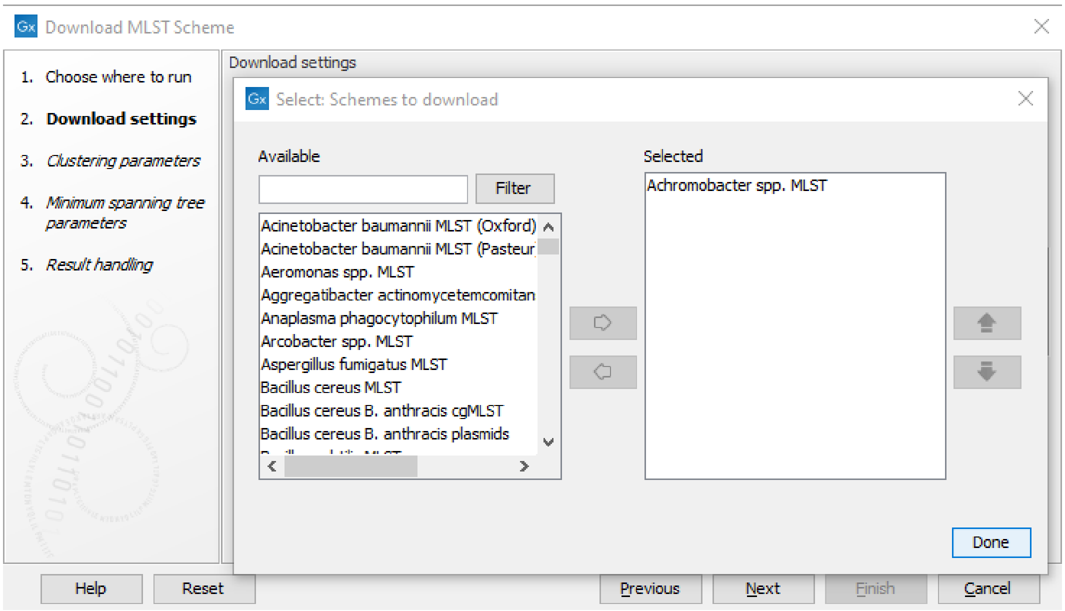
Use the **Schemes to download** selector (figure 13.6) to choose which schemes to download from PubMLST.

Most of the schemes offered for download by PubMLST are classic (7-gene) schemes, but there are also core genome schemes available for several species, e.g.: N. gonorrhoeae, N. Meningitis, C. Jejuni / C. Coli, C. trachomatis, Vibrio cholerae, Listeria monocytogenes.

Some of the schemes offered by PubMLST may only contain allele and locus definitions and no sequence types.

The **Download metadata** option makes it possible to download and extract metadata for all of the isolates for a given species in PubMLST. Note that this is a potentially very slow operation.

The clustering parameters determine how the heatmap should be clustered (figure 13.7). The heatmap cell values are the observed frequencies of a given allele compared to the other alleles in the same locus.The possible cluster linkages are:



1. Choose where to run
2. **Download settings**
3. *Clustering parameters*
4. *Minimum spanning tree parameters*
5. *Result handling*

Download settings

Available

Acinetobacter baumannii MLST (Oxford)
Acinetobacter baumannii MLST (Pasteur
Aeromonas spp. MLST
Aggregatibacter actinomycetemcomitan
Anaplasma phagocytophilum MLST
Arcobacter spp. MLST
Aspergillus fumigatus MLST
Bacillus cereus MLST
Bacillus cereus B. anthracis cgMLST
Bacillus cereus B. anthracis plasmids

Selected

Achromobacter spp. MLST

Figure 13.6: *The schemes available for download.*

- **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.
- **Complete linkage**: The distance between two clusters is computed as the distance between the two farthest objects in the two clusters.

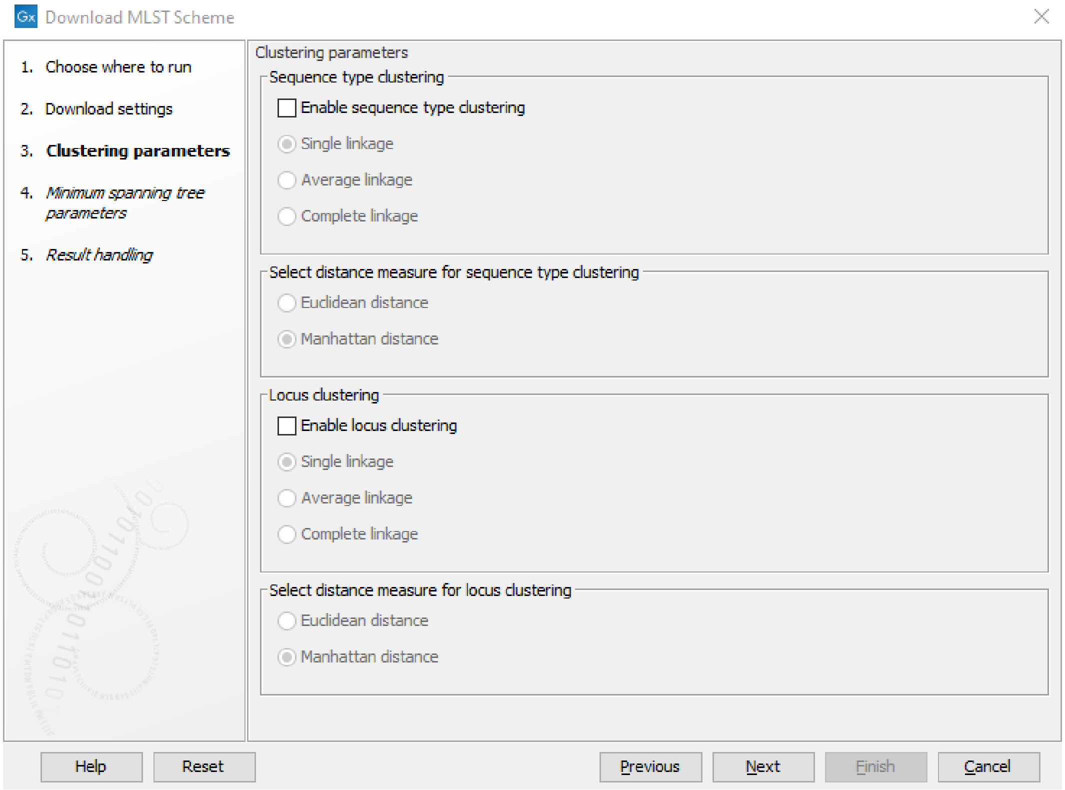
The possible distance measures are:

- **Euclidean distance**: the square-root of the sum-of-square differences between coordinates.
- **Manhattan distance**: the sum of absolute differences between coordinates.

Note that for schemes with thousands of sequence types and/or loci, the clustering may become very slow and time-consuming.

The following options are available when creating a minimum spanning tree (figure 13.8):

- **Comparing a known to a missing allele:** the minimum spanning tree is created using a distance matrix, where the distance is calculated between all pairs of sequence types. The distance is calculated as the number of loci where the allele assignment differs. But in some cases, a locus for a sequence type may not have an assigned allele (for instance, for the accessory genes in a wgMLST scheme). If this is the case, the behavior depends on this setting: if 'counted as same alleles' is selected, a locus where at least one allele is missing for the pair being compared will be ignored (it will not count as a difference). On the other hand, if 'Counted as different alleles' is selected, a missing allele being compared to a known allele will increase the distance between the sequence types being compared.



1. Choose where to run
2. Download settings
3. **Clustering parameters**
4. *Minimum spanning tree parameters*
5. *Result handling*

Clustering parameters

Sequence type clustering

☐ Enable sequence type clustering

● Single linkage

○ Average linkage

○ Complete linkage

Select distance measure for sequence type clustering

○ Euclidean distance

● Manhattan distance

Locus clustering

☐ Enable locus clustering

● Single linkage

○ Average linkage

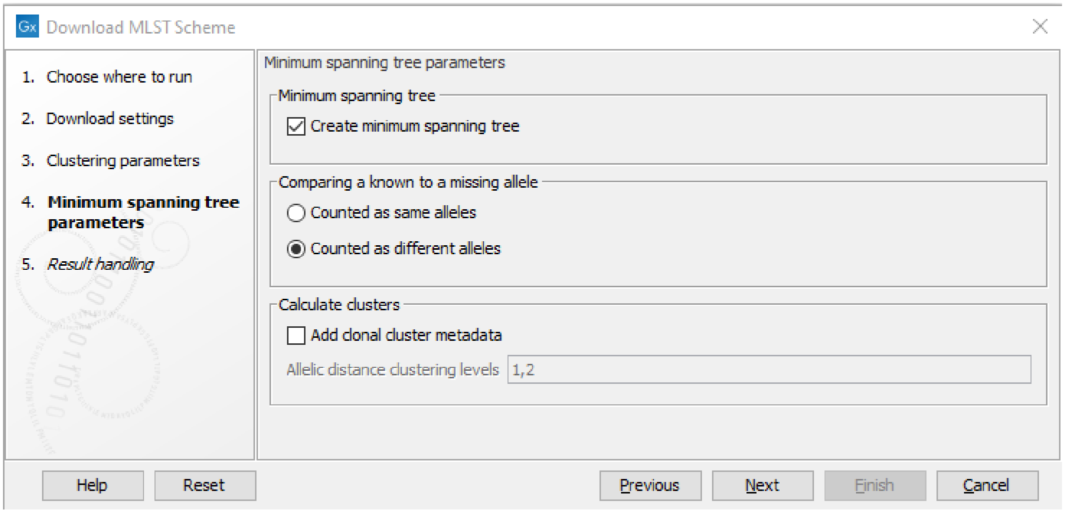
○ Complete linkage

Select distance measure for locus clustering

○ Euclidean distance

● Manhattan distance

Figure 13.7: *The clustering parameters.*



1. Choose where to run
2. Download settings
3. Clustering parameters
4. **Minimum spanning tree parameters**
5. *Result handling*

Minimum spanning tree parameters

Minimum spanning tree

☑ Create minimum spanning tree

Comparing a known to a missing allele

○ Counted as same alleles

● Counted as different alleles

Calculate clusters

☐ Add clonal cluster metadata

Allelic distance clustering levels 1,2

Figure 13.8: *The minimum spanning tree parameters.*

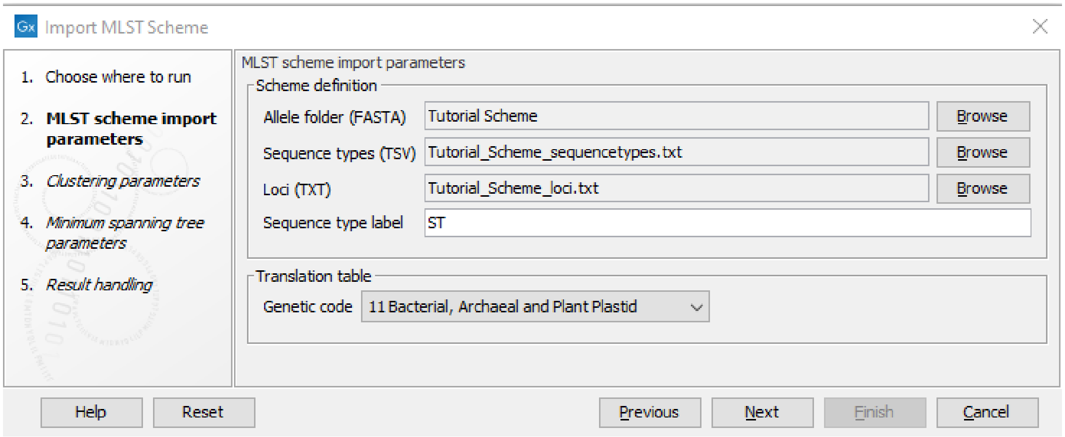
- **Add clonal cluster metadata:** it is possible to assign cluster information to the scheme which will show up as metadata. The clustering is based on the minimum spanning tree, and will be similar to the clustering obtained by using the 'collapse branches' slider in the minimum spanning tree view - that is, the clustering will be single-linkage clustering - i.e. all nodes in cluster are within the specified threshold to at least one other node in the cluster. Each cluster will get a name chosen from the sequence type in the cluster with most connections.

- **Add clonal cluster metadata:** specifies the level at which the clustering will be performed. It is possible to specify multiple, comma-separated values. E.g. ’100,200’ will assign clusters at allelic distances of 100 and 200 - this will create two new metadata columns, cc_100 and cc_200 with the new cluster information.

13.3 Import MLST Scheme

To run the Import MLST Scheme tool choose:

Toolbox | **Microbial Genomics Module** ( ) | **Databases** ( ) | **MLST Typing** ( ) | **Import MLST Scheme** ( )



Import MLST Scheme

1. Choose where to run
2. **MLST scheme import parameters**
3. *Clustering parameters*
4. *Minimum spanning tree parameters*
5. *Result handling*

MLST scheme import parameters

Scheme definition

Allele folder (FASTA): Tutorial Scheme

Sequence types (TSV): Tutorial_Scheme_sequencetypes.txt

Loci (TXT): Tutorial_Scheme_loci.txt

Sequence type label: ST

Translation table

Genetic code: 11 Bacterial, Archaeal and Plant Plastid

Figure 13.9: *The MLST Scheme import parameters.*

The **Allele folder (FASTA)** must contain a set of FASTA files for each locus. The files must have one of the following extensions to be recognized: "fa", "fas", "fsa", "fasta", "tfa". The name of the allele must be the locus name and allele name separated by an underscore, like in this example:

```
>pheS_1
AGAGAAAAGAACGATACTTTCTATATGGCCCGTGATAATCAAGGCAAGCGTGTTGTCTTA
>pheS_2
AGAGAAAAGAACGATACTTTCTATATGGCCCGTGATAATCAAGGCAAGCGTGTTGTCTTA
```

The **Sequence types (TSV)** file must be a tab-separated file listing a sequence type and its alleles in the following format:

```
ST  pheS    glyA    fumC    mdh sucA    dnaN    atpA    clonal_complex
1   30  1   1   1   1   1   1   6
3   6   8   7   3   4   3   1
4   7   9   8   3   5   2   1
6   47  3   10  4   7   2   2
```

It is possible to add arbitrary metadata as additional columns after the loci columns (e.g. the ’clonal_complex’ column above). If multiple isolates share the same sequence type, but have different metadata, it is possible to add multiple lines with the same sequence type name and allele ids, but with different metadata entries.

The **Loci (TXT)** file must be a tab-separated file listing a locus name and its corresponding metadata. For this file the only recognized headers are "Locus", "Known name", "Type name", "Locus type" where the name of the locus in the MLST scheme needs to match the name in the Locus column of the annotation file.

```
Locus Known name Type name Locus type
locus5 FALSE Unknown ST1
fliR TRUE fli ST2
flgL TRUE flg ST3
hpaB TRUE hpa ST4
```

The **Clustering parameters** and **Minimum Spanning Tree parameters** are similar to the options for Download MLST Scheme tool (see section 13.2)

The genetic code specified will be used for novel allele detection to make sure each allele starts and ends with an initiation and stop codon, respectively. If "No code specified" is selected, these requirements will not be checked when searching for novel alleles, instead the aligned part of the existing alleles to the assembly is used to define the allelic length. Note that the latter is useful for 7-gene MLST schemes which generally use fractions of genes, but it is also sensitive towards unaligned ends and may return too short alleles in some cases.

Chapter 14

Databases for Amplicon-Based Analysis

14.1 Download Amplicon-Based Reference Database

Amplicon-based reference databases contain a list of representative amplicon sequences and their taxonomy. Such database is required for amplicon-based analysis (chapter 4).

The following databases are available:

- **SILVA**. Small subunit (SSU; 16S/18S) and large subunit (LSU; 23S/28S) ribosomal RNA sequences for prokaryotic and eukaryotic taxonomic assignment [Quast et al., 2012] (https://www.arb-silva.de/no_cache/download/archive/current/Exports/).
- **MiDAS**. 16S ribosomal RNA sequences for prokaryotic and eukaryotic taxonomic assignment of microbes in wastewater treatment and bioenergy systems [Dueholm et al., 2022] (https://www.midasfieldguide.org/guide/downloads).
- **UNITE**. Internal transcribed spacer (ITS) sequences for fungal taxonomic assignment [Kõljalg et al., 2020] (https://unite.ut.ee/repository.php).
- **Greengenes2**. Full length 16S ribosomal RNA sequences from the backbone of the Greengenes2 phylogenetic tree. For prokaryotic taxonomic assignment [McDonald et al., 2022] (https://greengenes2.ucsd.edu/).

Some of the above databases are available at different clustering levels of sequence similarities.

To run the tool, go to

> **Toolbox** | **Microbial Genomics Module** ( ) | **Metagenomics** ( ) | **Databases** ( ) | **Amplicon-Based Analysis** ( ) | **Download Amplicon-Based Reference Database** ( )

Select the database needed and specify where to save it. When using this tool, the databases downloaded are automatically formatted.

If you wish to format your own database with your own sequences and a corresponding taxonomy file, use the **Update Sequence Attributes in Lists** tool (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Update_Sequence_Attributes_in_Lists.html) to set the "Taxonomy" field. A clustering level for such custom databases can not be set on the data object directly, but it may be specified as a parameter when running the OTU Clustering tool.

Chapter 15

Databases for Taxonomic Analysis

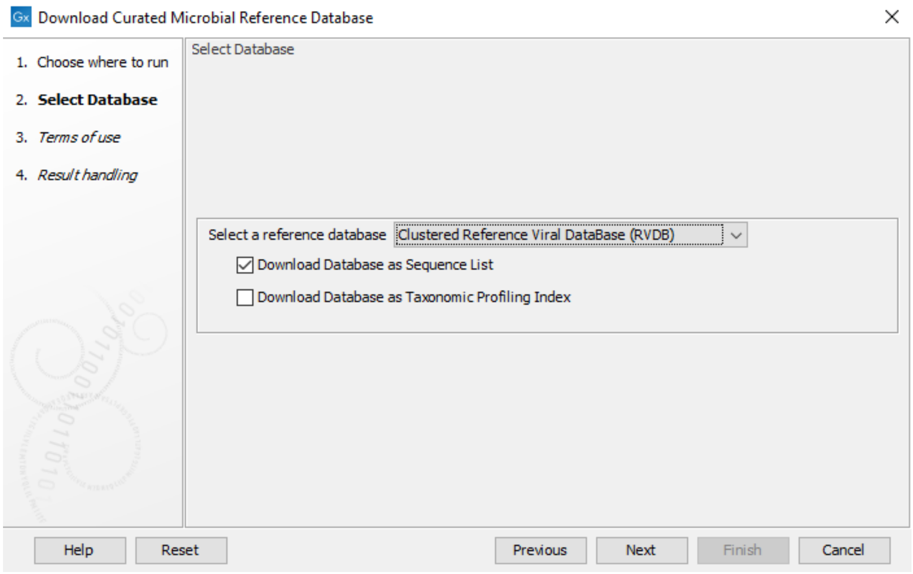
15.1 Download Curated Microbial Reference Database

The **Download Curated Microbial Reference Database** tool downloads selected references as sequence lists and/or taxonomic profiling indexes with the necessary annotations required for the tools in the Typing and Epidemiology and Metagenomics sections of the Microbial Genomics Module.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** () | **Metagenomics** () | **Databases** () | **Taxonomic Analyses** () | **Download Curated Microbial Reference Database** ()

In the first window (figure 15.1), select the database you wish to download.



Select a reference database: Clustered Reference Viral DataBase (RVDB)

☑ Download Database as Sequence List

□ Download Database as Taxonomic Profiling Index

Figure 15.1: *Select the database and output format*

You can choose between several databases

- **QMI-PTDB Genus.** QIAGEN Microbial Insights - Prokaryotic Taxonomy Database is a

microbial reference database for taxonomic profiling of bacteria and archaea. The database represents all genera with a varying number of species per genus.
Genome sequences and annotations are from the NCBI Reference Sequence Database (RefSeq; https://www.ncbi.nlm.nih.gov/refseq/) and have been annotated with taxonomy from the Genome Taxonomy Database (GTDB; https://gtdb.ecogenomic.org).
The database was created by selecting one representative genome per species, and subsequently reducing the relative number of species per genus to meet the desired database size. For reduction, higher assembly status, lower number of contigs, and longer total length was prioritized. All genomes marked as "reference genome" were retained. So were species commonly included in microbial reference standards.
When running Taxonomic Profiling with the QMI-PTDB Genus database, 32GB of memory is required.

- **QMI-PTDB Family**. QIAGEN Microbial Insights - Prokaryotic Taxonomy Database is a microbial reference database for taxonomic profiling of bacteria and archaea. The database represents all families with a varying number of genera per family.
Genome sequences and annotations are from the NCBI Reference Sequence Database (RefSeq; https://www.ncbi.nlm.nih.gov/refseq/) and have been annotated with taxonomy from the Genome Taxonomy Database (GTDB; https://gtdb.ecogenomic.org).
The database was created by selecting one representative genome per genus, and subsequently reducing the relative number of genera per family to meet the desired database size. For reduction, higher assembly status, lower number of contigs, and longer total length was prioritized. All genomes marked as "reference genome" were retained. So were species commonly included in microbial reference standards.
When running Taxonomic Profiling with the QMI-PTDB Family database, 16GB of memory is recommended.

- **Unified Human Gastrointestinal Genome (UHGG)**. A database of metagenomic-assembled genomes from human gut samples, curated and hosted by MGnify [Gurbich et al., 2023], EMBL-EBI (https://www.ebi.ac.uk/metagenomics/browse/genomes).

- **Unclustered Reference Viral DataBase (U-RVDB)**. Unclustered Reference Viral Database for virus detection [Goodacre et al., 2018]. The database includes curated viral, virus-related and virus-like nucleotide sequences except bacterial viruses, which are excluded.

- **Clustered Reference Viral DataBase (C-RVDB)**. Clustered Reference Viral Database for virus detection [Goodacre et al., 2018]. The database includes curated viral, virus-related and virus-like nucleotide sequences except bacterial viruses, clustered at 98% sequence similarity.

- **ViraCura^{TM} HPV REF**. A curated database of Human Papillomavirus reference strains. It contains unmodified viral reference genomes and associated record information from NCBI databases.

- **ViraCura^{TM} HPV VAR**. A curated database of Human Papillomavirus variants of reference strains. It contains unmodified viral reference genomes and associated record information from NCBI databases.

- **ViraCura^{TM} ANIMAL PV**. A curated database of Animal Papillomavirus. It contains unmodified viral reference genomes and associated record information from NCBI databases.

- **MPXV**. A curated database of Monkeypox virus reference strains. It contains unmodified viral reference genomes and associated record information from NCBI databases, as well as metadata and customized taxonomic nomenclature.
- **MOCOVA**. A curated database of Monkeypox outgroup reference strains (Molluscum contagiosum, Cowpox, Variola, and Vaccinia). It contains unmodified viral reference genomes and associated record information from NCBI databases, as well as metadata and customized taxonomic nomenclature.

You can then chose to download the database as an annotated sequence list and/or as a taxonomic profiling index.

The **Curated Microbial Reference Databases** are optimized for balance in the taxonomic representation across the taxonomy, i.e. the oversampling of some branches of the taxonomy is removed by using representative sequences. This has the consequence that some assemblies may not be particularly good assemblies, yet they are included as they constitute the best current representative of the given branch in the taxonomy. For this optimized database you can choose to download the 22g database, or one that is optimized for running the Taxonomic Profiling tool on a laptop computer with 16GB of main memory. The 16g version of the curated database contain a smaller number of assemblies, in order to be able to run on a system with 16GB of main memory.

Note: some of the databases offered are derived works, licensed under a Creative Commons Attribution-ShareAlike (CC BY-SA) license. We offer free access to those without requiring a CLC product license. They can be downloaded using the CLC Genomics Workbench with the Microbial Genomics Module installed in viewing mode. The downloaded files can then be exported to non-proprietary formats using the freely available viewing mode of the CLC Genomics Workbench.

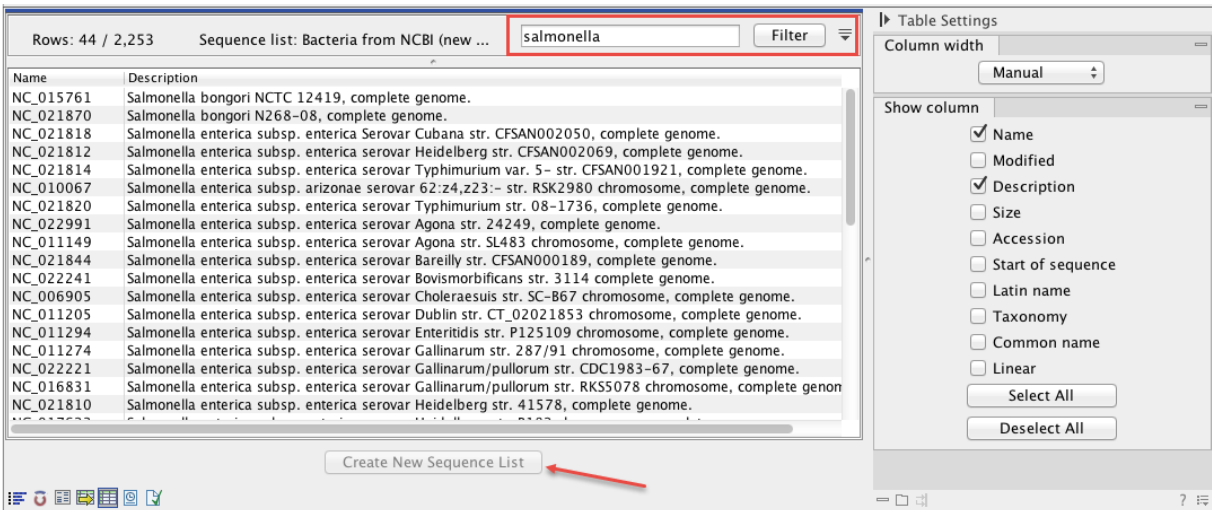
15.1.1 Extracting a subset of a database

After download, it is possible to select a subset of sequences and saving the reduced list in a new sequence list. This can reduce subsequent analysis runtime significantly.

For example, from a collection of bacterial genomes that include multiple representatives of each genus, you can extract a genus specific subset of sequences to a new list:

1. **Open** the downloaded bacterial genomes database.
2. **Switch** to tabular element mode (▦).
3. **Filter** towards the desired genus (figure 15.2).
4. **Select** all remaining rows.
5. **Click** the **Create New Sequence List** button.
6. **Save** the subset reference list.

Another way to extract a subset of a database is to make use of the **Split Sequence List** tool. For more information, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Split_Sequence_List.html.



Rows: 44 / 2,253 Sequence list: Bacteria from NCBI (new ...

salmonella

Name	Description
NC_015761	Salmonella bongori NCTC 12419, complete genome.
NC_021870	Salmonella bongori N268-08, complete genome.
NC_021818	Salmonella enterica subsp. enterica Serovar Cubana str. CFSAN002050, complete genome.
NC_021812	Salmonella enterica subsp. enterica serovar Heidelberg str. CFSAN002069, complete genome.
NC_021814	Salmonella enterica subsp. enterica serovar Typhimurium var. 5- str. CFSAN001921, complete genome.
NC_010067	Salmonella enterica subsp. arizonae serovar 62:z4,z23:- str. RSK2980 chromosome, complete genome.
NC_021820	Salmonella enterica subsp. enterica serovar Typhimurium str. 08-1736, complete genome.
NC_022991	Salmonella enterica subsp. enterica serovar Agona str. 24249, complete genome.
NC_011149	Salmonella enterica subsp. enterica serovar Agona str. SL483 chromosome, complete genome.
NC_021844	Salmonella enterica subsp. enterica serovar Bareilly str. CFSAN000189, complete genome.
NC_022241	Salmonella enterica subsp. enterica serovar Bovismorbificans str. 3114 complete genome.
NC_006905	Salmonella enterica subsp. enterica serovar Choleraesuis str. SC-B67 chromosome, complete genome.
NC_011205	Salmonella enterica subsp. enterica serovar Dublin str. CT_02021853 chromosome, complete genome.
NC_011294	Salmonella enterica subsp. enterica serovar Enteritidis str. P125109 chromosome, complete genome.
NC_011274	Salmonella enterica subsp. enterica serovar Gallinarum str. 287/91 chromosome, complete genome.
NC_022221	Salmonella enterica subsp. enterica serovar Gallinarum/pullorum str. CDC1983-67, complete genome.
NC_016831	Salmonella enterica subsp. enterica serovar Gallinarum/pullorum str. RKS5078 chromosome, complete genon
NC_021810	Salmonella enterica subsp. enterica serovar Heidelberg str. 41578, complete genome.

Table Settings

Column width: Manual

Show column

- ☑ Name
- ☐ Modified
- ☑ Description
- ☐ Size
- ☐ Accession
- ☐ Start of sequence
- ☐ Latin name
- ☐ Taxonomy
- ☐ Common name
- ☐ Linear

Figure 15.2: *The downloaded NCBI bacterial genomes database was filtered for Salmonella data. A subset of 44 out of 2,253 sequences matched this search criterion.*

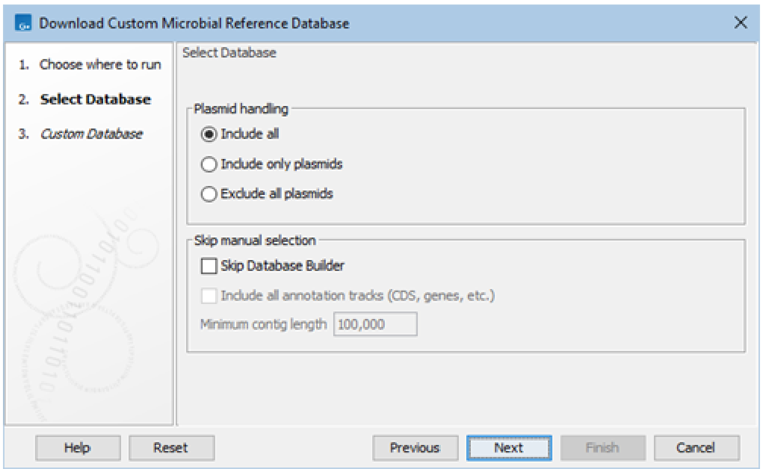
15.2 Download Custom Microbial Reference Database

The Download Custom Microbial Reference Database tool allows you to create a custom database from taxonomies or NCBI assembly IDs. The tool outputs a single sequence list.

To run the tool, go to:

Toolbox | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Taxonomic Analyses** ( ) | **Download Custom Microbial Reference Database** ( )

Under **Customize Database**, select whether to include genomic and/or plasmid sequences (figure 15.3):



Download Custom Microbial Reference Database

1. Choose where to run
2. **Select Database**
3. *Custom Database*

Select Database

Plasmid handling

- ● Include all
- ○ Include only plasmids
- ○ Exclude all plasmids

Skip manual selection

- ☐ Skip Database Builder
- ☐ Include all annotation tracks (CDS, genes, etc.)

Minimum contig length: 100,000

Figure 15.3: *Select type of sequences to include and whether to skip the Database Builder.*

- **Include all**. The database will contain both genomic and plasmid sequences.
- **Include only plasmids**. The database will contain only plasmid sequences.

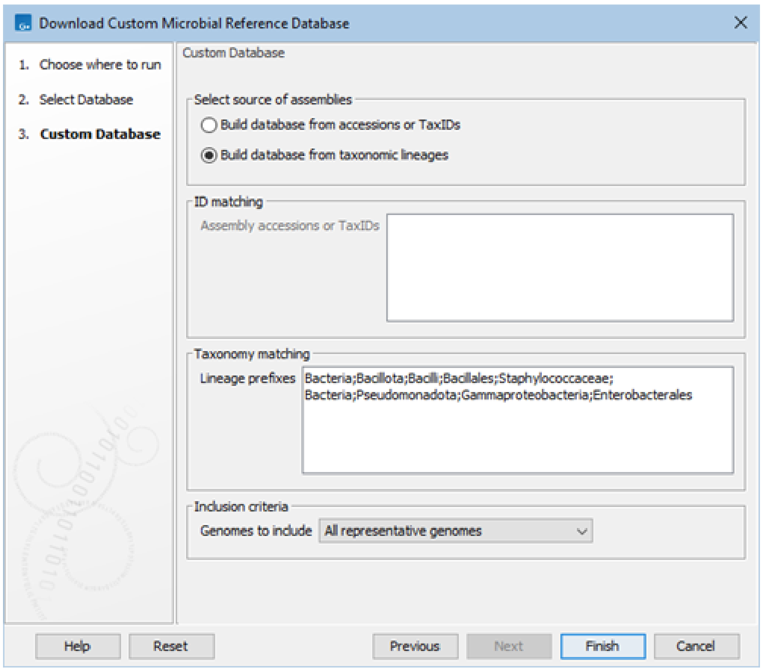
- **Exclude all plasmids**. The database will not contain any plasmid sequences.

Choose whether you wish to skip manual selection:

- **Skip Database Builder**. If checked, a reference database with genomes matching the specified criteria will be downloaded once you click **Finish** from the next wizard step.

  If left unchecked, clicking **Finish** will instead open the **Database Builder** from which you can manually select genomes for download, see section 15.2.1. Genomes that match the specified criteria will be pre-selected.
- **Include all annotation tracks**. Will include CDS, gene, etc. annotations in the downloaded database. The annotations are not needed for taxonomic profiling, but may be required for other applications such as creating MLST schemes.
- **Minimum contig length**. The minimum length of sequences to be included in the database.

Click **Next** to customize the database (figure 15.4):



Download Custom Microbial Reference Database

1. Choose where to run
2. Select Database
3. **Custom Database**

Custom Database

Select source of assemblies
○ Build database from accessions or TaxIDs
● Build database from taxonomic lineages

ID matching
Assembly accessions or TaxIDs

Taxonomy matching
Lineage prefixes: Bacteria;Bacillota;Bacilli;Bacillales;Staphylococcaceae;
Bacteria;Pseudomonadota;Gammaproteobacteria;Enterobacterales

Inclusion criteria
Genomes to include: All representative genomes

Figure 15.4: *Specify accession or TaxIDs, or taxonomic lineages to include in the database.*

- **Select source of assemblies**:
  - **Build database from accessions or TaxIDs**. Enables the **ID matching** field, see below.
  - **Build database from taxonomic lineages**. Enables the **Taxonomic matching**, see below.
- **ID matching**. Provide a list of GenBank or RefSeq assembly accessions, or NCBI TaxIDs or species TaxIDs (one per line) to be included in your database.

If using GenBank or RefSeq assembly accessions, the accessions must follow the assembly accession: 3 letter prefix, (GCA for GenBank assemblies or GCF for RefSeq assemblies) followed by an underscore and 9 digits. For example, GCA_000019425 for the assembly of the DH10B substrain of E. coli. If a version number is included, it will be ignored and the newest version downloaded. The assembly is always downloaded from GenBank.

The TaxID is the NCBI taxonomy identifier for the organism from which the genome assembly was derived. The species TaxID is the identifier for the species to which the organism belongs. For a given organism, TaxID and species TaxID will be identical unless the organism was reported at a strain or subspecies level.

- **Taxonomy matching**. Provide a list of taxonomic lineage prefixes (one per line) to include in your database. The lineages should follow the format of 7-step taxonomies. For example entering "Bacteria;Bacillota;Bacilli;Bacillales;Staphylococcaceae;" will include all genera and species genomes in the Staphylococcaceae family. Entering "Bacteria;Pseudomonadota;Gammaproteobacteria;Enterobacterales;" will include all family, genera and species genomes in the Enterobacterales order. The NCBI taxonomy is updated weekly. When searching you should use the updated taxonomy.

- **Inclusion criteria**:

  - **All reference genomes**. All reference genomes in the chosen lineage(s) are included.
  - **All representative genomes**. All representative genomes in the chosen lineage(s) are included.
  - **All reference and representative genomes**. All reference and representative genomes in the chosen lineage(s) are included.
  - **All genomes**. All genomes in the chosen lineage(s) are included.
  - **One per species**. One reference is selected for each species in the chosen lineage(s). The chosen species representative is selected based on ranking with Reference genomes > Representative genomes > Complete genomes > Scaffolds > Contigs. When two or more references share the same rank, the reference with the longest chromosome is selected. Note species are identified using species TaxIDs. This means that assemblies with different species names but the same species TaxIDs are considered as one species.
  - **One per genus**. One reference is selected for each genus in the chosen lineage(s). The chosen genus representative is selected based on ranking with Reference genomes > Representative genomes > Complete genomes > Scaffolds > Contigs. When two or more references share the same rank, the reference with the longest chromosome is selected.

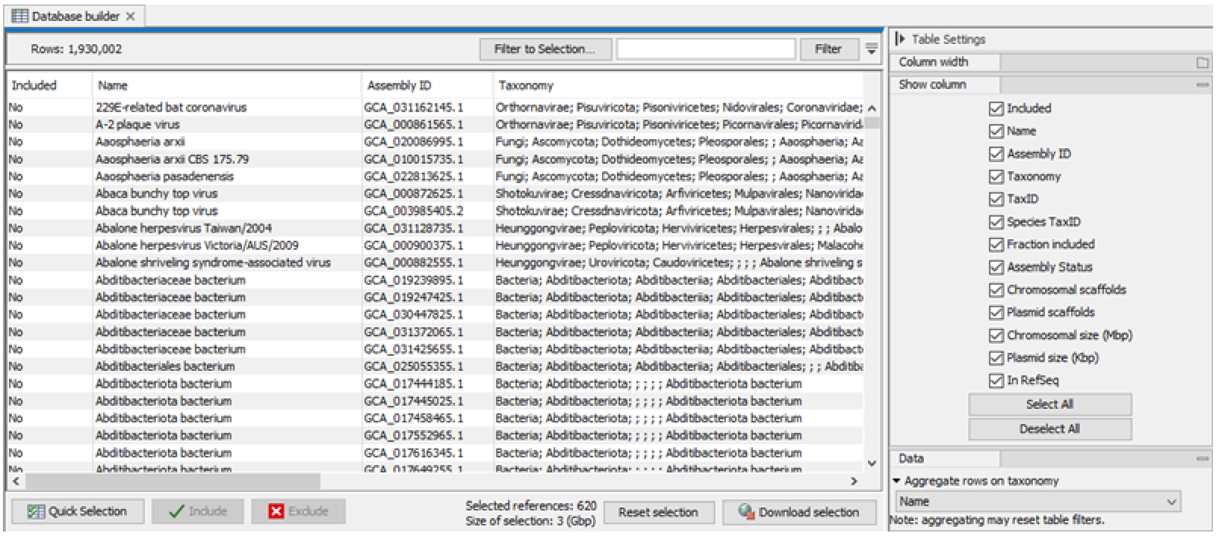
Click **Finish**.

If **Skip Database Builder** was selected, all genomes matching the specified criteria will now be downloaded. If the enabled ID or Taxonomy matching field was left empty, no genomes will be downloaded.

If **Skip Database Builder** was left unchecked, a reference database is not downloaded right away. Instead, the Database Builder will open, see section 15.2.1.

15.2.1 Database Builder

Depending on your internet connection, it takes a few seconds to download the content and open the **Database Builder** (figure 15.5).



Rows: 1,930,002

Included	Name	Assembly ID	Taxonomy
No	229E-related bat coronavirus	GCA_031162145.1	Orthornavirae; Pisuviricota; Pisoniviricetes; Nidovirales; Coronaviridae;
No	A-2 plaque virus	GCA_000861565.1	Orthornavirae; Pisuviricota; Pisoniviricetes; Picornavirales; Picornavirid
No	Aaosphaeria arxii	GCA_020086995.1	Fungi; Ascomycota; Dothideomycetes; Pleosporales; ; Aaosphaeria; Aa
No	Aaosphaeria arxii CBS 175.79	GCA_010015735.1	Fungi; Ascomycota; Dothideomycetes; Pleosporales; ; Aaosphaeria; Aa
No	Aaosphaeria pasadenensis	GCA_022813625.1	Fungi; Ascomycota; Dothideomycetes; Pleosporales; ; Aaosphaeria; Aa
No	Abaca bunchy top virus	GCA_000872625.1	Shotokuvirae; Cressdnaviricota; Arfiviricetes; Mulpavirales; Nanovirida
No	Abaca bunchy top virus	GCA_003985405.2	Shotokuvirae; Cressdnaviricota; Arfiviricetes; Mulpavirales; Nanovirida
No	Abalone herpesvirus Taiwan/2004	GCA_031128735.1	Heunggongvirae; Peploviricota; Herviviricetes; Herpesvirales; ; ; Abalo
No	Abalone herpesvirus Victoria/AUS/2009	GCA_000900375.1	Heunggongvirae; Peploviricota; Herviviricetes; Herpesvirales; Malacohe
No	Abalone shriveling syndrome-associated virus	GCA_000882555.1	Heunggongvirae; Uroviricota; Caudoviricetes; ; ; ; Abalone shriveling s
No	Abditibacteriaceae bacterium	GCA_019239895.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; Abditibact
No	Abditibacteriaceae bacterium	GCA_019247425.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; Abditibact
No	Abditibacteriaceae bacterium	GCA_030447825.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; Abditibact
No	Abditibacteriaceae bacterium	GCA_031372065.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; Abditibact
No	Abditibacteriaceae bacterium	GCA_031425655.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; Abditibact
No	Abditibacteriales bacterium	GCA_025055355.1	Bacteria; Abditibacteriota; Abditibacteriia; Abditibacteriales; ; ; Abditib
No	Abditibacteriota bacterium	GCA_017444185.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium
No	Abditibacteriota bacterium	GCA_017445025.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium
No	Abditibacteriota bacterium	GCA_017458465.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium
No	Abditibacteriota bacterium	GCA_017552965.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium
No	Abditibacteriota bacterium	GCA_017616345.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium
No	Abditibacteriota bacterium	GCA_017649255.1	Bacteria; Abditibacteriota; ; ; ; ; Abditibacteriota bacterium

Selected references: 620
Size of selection: 3 (Gbp)

☑ Included
☑ Name
☑ Assembly ID
☑ Taxonomy
☑ TaxID
☑ Species TaxID
☑ Fraction included
☑ Assembly Status
☑ Chromosomal scaffolds
☑ Plasmid scaffolds
☑ Chromosomal size (Mbp)
☑ Plasmid size (Kbp)
☑ In RefSeq

Aggregate rows on taxonomy: Name

Note: aggregating may reset table filters.

Figure 15.5: *Search, filter and select assemblies to download.*

Assemblies that match the criteria from the Download Custom Microbial Reference Database tool will be pre-selected, indicated by a "Yes" in the *Included* column.

The **Database Builder** table contains additional columns with metadata based on information from GenBank, https://www.ncbi.nlm.nih.gov/genbank/. Use the **Database Builder** functionality described below to customize and define the reference set to be downloaded.

Use the filtering options located at the top right to filter the table. For information on how to use the simple and advanced table filters, see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Filtering_tables.html.

From the **Side Panel** on the right, the following option is available:

- **Aggregate rows on taxonomy**. Aggregates results by the selected taxonomic level, e.g. *Order*.

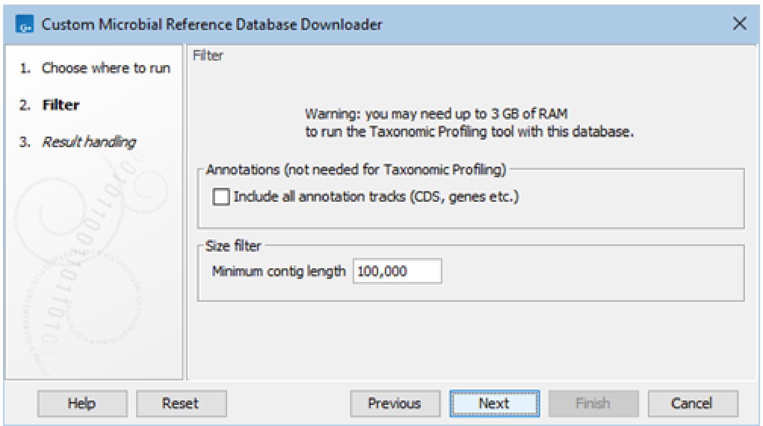
Below the table you find buttons for quick selection, including or excluding rows, and download of selected reference subset:

- **Quick selection**. For selection of one of the following predefined subsets, based on information in the *Assembly Status*, *Chromosomal scaffolds*, and *In RefSeq* columns:
  - **Single scaffold complete genomes in RefSeq**. Complete genomes with *Chromosomal scaffolds*= 1; *In RefSeq*= Yes.
  - **Complete genomes in RefSeq**. Complete genomes with *In RefSeq*= Yes.
  - **All complete genomes**. Any Complete genome.

  "Complete genome" refers to the *Assembly Status*. All genomes marked as Complete genome or Chromosome are included in the subsets, as are any reference marked as representative genome (repr) or reference genomes (refr).

- **Include** and **Exclude**. Includes or excludes the selected rows from the subset selection.
- **Reset selection**. Reset selection to match criteria specified in **Download Custom Microbial Reference Database** wizard.
- **Download selection**. For download of the selected reference subset. Brings up a dialog with the following options (figure 15.6):
  - **Include all annotation tracks**. Will include CDS, gene, etc. annotations in the downloaded database. The annotations are not needed for taxonomic profiling, but may be required for other applications such as creating MLST schemes.
  - **Minimum contig length**. The minimum length of sequences to be included in the database.

    The dialog provides an estimate of the memory and disk requirements needed to later run the Taxonomic Profiling tool with the database you are about to download.



Custom Microbial Reference Database Downloader

1. Choose where to run
2. **Filter**
3. *Result handling*

Filter

Warning: you may need up to 3 GB of RAM
to run the Taxonomic Profiling tool with this database.

Annotations (not needed for Taxonomic Profiling)

□ Include all annotation tracks (CDS, genes etc.)

Size filter

Minimum contig length 100,000

Figure 15.6: *Filter options for download of the selected references.*

15.3 Download Pathogen Reference Database

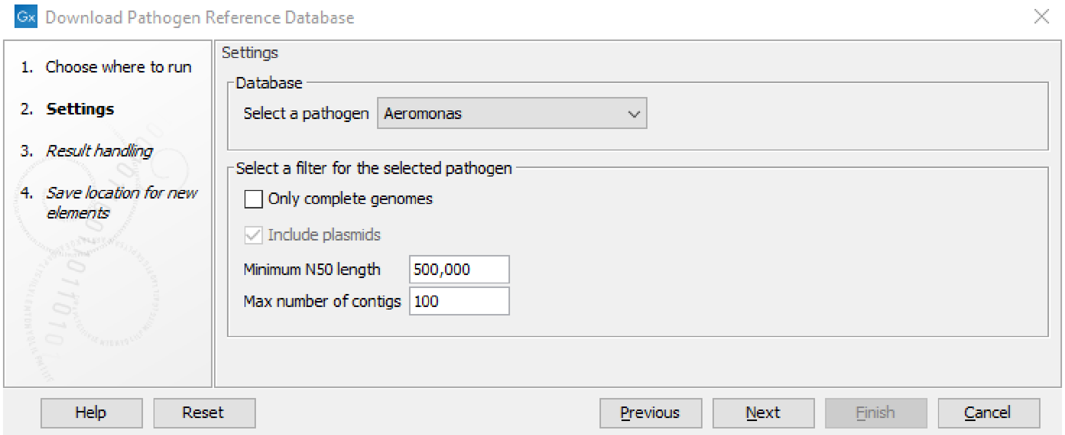
Download a collection of bacterial assemblies and enrich with metadata from the NCBI Pathogen Detection Project (see https://www.ncbi.nlm.nih.gov/projects/pathogens/).

Toolbox | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Taxonomic Analyses** ( ) | **Download Pathogen Reference Database** ( )

This will open the following wizard window (figure 15.7):

The settings are:

- **Select a pathogen**. Select a pathogen for which to download assemblies and associated metadata.
- **Only complete genomes**. This can be used to switch between complete genomes or to also allow for downloading incomplete assemblies.
- **Include plasmids**. This option can be used to include or exclude plasmids from the downloaded database. Note that if a database of plasmids only is required, the Download Custom Microbial Reference Database tool should be used instead.



Download Pathogen Reference Database

1. Choose where to run
2. **Settings**
3. *Result handling*
4. *Save location for new elements*

Settings

Database

Select a pathogen: Aeromonas

Select a filter for the selected pathogen

□ Only complete genomes

☑ Include plasmids

Minimum N50 length: 500,000

Max number of contigs: 100

Figure 15.7: *Downloading assemblies and metadata for a selected pathogen from the NCBI Pathogen Detection Project.*

- **Minimum N50 length.** This option can be used to remove assemblies with shorter N50 values (the default value is set at 500,000 bp). Short N50 values typically indicate low assembly quality. This option is not available when "Only complete genomes" has been selected.

- **Maximum number of contigs**. This option can be used to remove assemblies with a higher number of contigs (the default value is set at 100). Many contigs typically indicate low assembly quality. This option is not available when "Only complete genomes" has been selected,

Specify a location to save the database. We recommend to create a folder where you can save all the databases and MLST schemes necessary to run some of the CLC Microbial Genomics Module tools.

The resulting database includes a list of different bacterial genome sequences as well as the associated accession numbers, descriptions, taxonomy and size of the sequences. In addition, each reference genome will be annotated with the following metadata (when available):

- serovar
- strain
- taxonomy
- sample collection date
- geographical location
- isolation source
- host
- host disease
- outbreak
- SRA run id

- SRA project id

Once a database has been downloaded, it is possible to extract a subset following the instructions described in section 15.1.1.

15.4 Create Taxonomic Profiling Index

This tool will compute a a taxonomic profiling index from a reference database. Indexes are then used as input to the Taxonomic Profiling tool. The computation of index files for taxonomic profiling is memory and hard-disk intensive due to the large sizes of reference databases usually employed for this task. The algorithm requires roughly the number of bases in bytes of memory (as indicated by the Download Custom Microbial Reference Database tool), i.e., approximatively the size of the uncompressed reference database; and twice this amount in hard disk space.

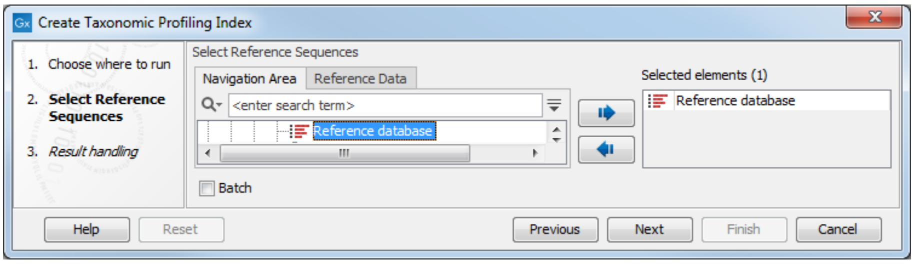


Figure 15.8: *Select sequence lists with the references of interest.*

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Taxonomic Analysis** | **Create Taxonomic Profiling Index** ( )

As input, select one or more sequence lists containing the references of interest. These can be downloaded for example using **Download Custom Microbial Reference Database** (section 15.2).

The tool makes use of Assembly IDs (see section 21) in combination with either Latin name or, if Latin name is not present, Sequence name. The tool will treat sequences as one reference, if they have:

- Identical Assembly ID *and* same Latin name, or
- Identical Assembly ID *and* same unique sequence name

The output is an index file and a report as seen in figure 15.9. The report list the number of sequence and basepairs that were indexed.

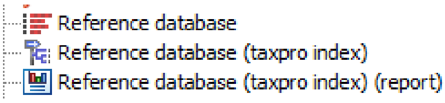

Reference database
Reference database (taxpro index)
Reference database (taxpro index) (report)

Figure 15.9: *The reference sequences,index and report as seen in the Navigation Area.*

Chapter 16

Databases for Functional Analysis

16.1 Download Protein Database

The **Download Protein Database** allows you to download the following protein databases:

- **Clusters of Orthologous Genes (COG)**
- **SwissPROT (with GO-term annotations)**
- **UniProt (UniRef50 complete with GO & EC annotations)**. Entries are annotated with Enzyme Commission (EC) numbers and GO-terms where available. Notice, that entries with GO-terms and no EC terms are not included.
- **UniProt (UniRef90 subset with GO & EC annotations)**. This is the subset of entries containing Enzyme Commission (EC) numbers and GO terms. Notice, that entries with GO-terms and no EC terms are not included.

These protein databases can be used to create DIAMOND Indexes (see section 16.4), that may be used together with the Annotate CDS with Best DIAMOND Hit tool (see section 11.5) and Annotate with DIAMOND tool (see section 11.3).

Notice, that the **SwissPROT** and **UniProt (UniRef50)** protein reference databases have been annotated with GO associations from the EBI Gene Ontology Annotation (GOA) Database (`https://www.ebi.ac.uk/GOA/index`).

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** () | **Databases** () | **Functional Analysis** () | **Download Protein Database** ()

Choose the database you wish to download from the drop-down menu, and when needed, accept the terms of use before clicking Finish to save the database in the Navigation Area.

16.2 Download Ontology Database

The **Download Ontology Database** tool allows you to download the latest versions of:

- the GO database and of Pfam2GO mappings from the Gene Ontology Consortium (https://geneontology.org/).
- the Enzyme Commission number database from Expasy (https://enzyme.expasy.org/).

The GO database is used to convert Pfam annotation to GO terms by the Annotate CDS with Pfam Domains tool (see section 11.6) and by the Build Functional Profile tool (see section 11.7).

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Functional Analysis** ( ) | **Download Ontology Database** ( )

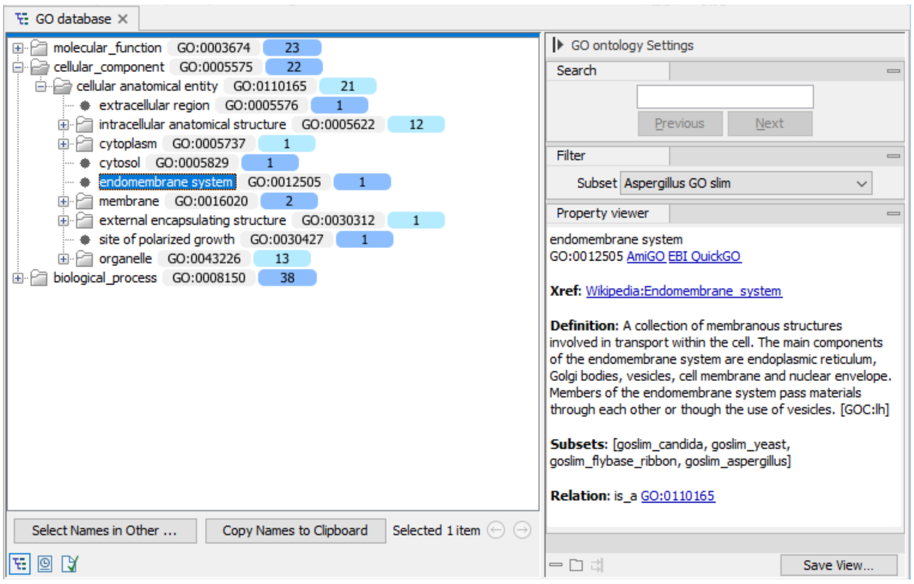
If you select **Create Report**, the tool will also generate a summary report table. For each downloaded file, the table will contain the name of the downloaded file, its size, the URL from which is was downloaded, and the number of entries in the file.

16.2.1 The GO Database View

When downloading the GO database, a new object called *GO database* ( ) is created in the Navigation Area.

The GO Database element has a default hierarchical tree view (See figure 16.1). Here it is possible to search the different ontology terms, and see their relations.

It should be noted that the Gene Ontology is not a tree, but a directed acyclic graph - this means that a GO-term may have multiple parents, and thus appear different places in the tree view. Likewise, when searching for a specific GO-term, multiple locations in the tree might be highlighted, even though they refer to the same GO-term.



molecular_function GO:0003674 23
cellular_component GO:0005575 22
cellular anatomical entity GO:0110165 21
extracellular region GO:0005576 1
intracellular anatomical structure GO:0005622 12
cytoplasm GO:0005737 1
cytosol GO:0005829 1
endomembrane system GO:0012505 1
membrane GO:0016020 2
external encapsulating structure GO:0030312 1
site of polarized growth GO:0030427 1
organelle GO:0043226 13
biological_process GO:0008150 38

Subset Aspergillus GO slim

endomembrane system
GO:0012505 AmiGO EBI QuickGO

Xref: Wikipedia:Endomembrane_system

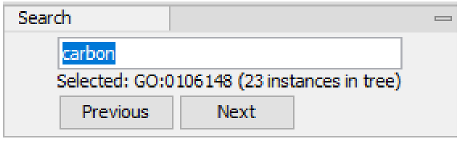
Definition: A collection of membranous structures involved in transport within the cell. The main components of the endomembrane system are endoplasmic reticulum, Golgi bodies, vesicles, cell membrane and nuclear envelope. Members of the endomembrane system pass materials through each other or though the use of vesicles. [GOC:lh]

Subsets: [goslim_candida, goslim_yeast, goslim_flybase_ribbon, goslim_aspergillus]

Relation: is_a GO:0110165

Figure 16.1: *The GO Database View*

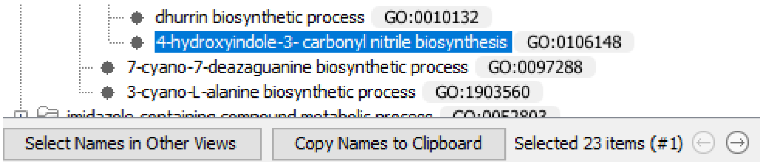
In figure 16.2 an example of a search result is shown. Here, there are multiple GO terms matching the search query 'carbon'. The **Previous** and **Next** buttons can be used to navigate to the matching GO terms. Notice, that this particular GO-term (GO:0106148) is shown multiple times in the tree ('23 instances in tree'). In order to view the different locations in the tree for a matching GO-term, the arrow buttons at the bottom of the view can be used to focus on the different selected elements (see figure 16.3) - note, that this does not change the selection, it only changes the focused area of the tree.



carbon

Selected: GO:0106148 (23 instances in tree)

Figure 16.2: *Searching for GO terms.*



dhurrin biosynthetic process GO:0010132
4-hydroxyindole-3- carbonyl nitrile biosynthesis GO:0106148
7-cyano-7-deazaguanine biosynthetic process GO:0097288
3-cyano-L-alanine biosynthetic process GO:1903560
imidazole-containing compound metabolic process

Selected 23 items (#1)

Figure 16.3: *Scrolling through different selected items in the GO view.*

The **Filter** sidepanel section, can be used to restrict to the view to various subsets of the full database ("Slim" subsets). It is also possible to restrict the tree to only the terms that are currently selected.

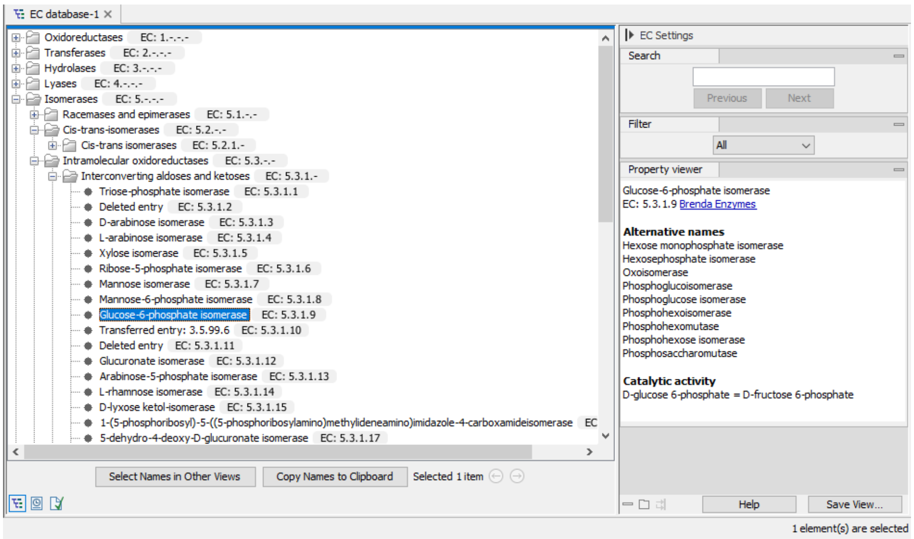
When clicking on a GO term, the Property viewer in the side panel will show the description, relations, synonyms, and all related links.

It is possible to **Select Names in Other Views** and **Copy Names to Clipboard** (see figure 16.3). Selecting names in other views will match names in other editors that support this - currently, this is only supported in the Differential Abundance element view.

16.2.2 The EC Database View

When downloading the EC database, a new object called *EC database* ( ) is created in the Navigation Area.

Similar to the GO database, the EC database has a hierarchical tree view (See figure 16.4).



Oxidoreductases EC: 1.-.-.-
Transferases EC: 2.-.-.-
Hydrolases EC: 3.-.-.-
Lyases EC: 4.-.-.-
Isomerases EC: 5.-.-.-
Racemases and epimerases EC: 5.1.-.-
Cis-trans-isomerases EC: 5.2.-.-
Cis-trans isomerases EC: 5.2.1.-
Intramolecular oxidoreductases EC: 5.3.-.-
Interconverting aldoses and ketoses EC: 5.3.1.-
Triose-phosphate isomerase EC: 5.3.1.1
Deleted entry EC: 5.3.1.2
D-arabinose isomerase EC: 5.3.1.3
L-arabinose isomerase EC: 5.3.1.4
Xylose isomerase EC: 5.3.1.5
Ribose-5-phosphate isomerase EC: 5.3.1.6
Mannose isomerase EC: 5.3.1.7
Mannose-6-phosphate isomerase EC: 5.3.1.8
Glucose-6-phosphate isomerase EC: 5.3.1.9
Transferred entry: 3.5.99.6 EC: 5.3.1.10
Deleted entry EC: 5.3.1.11
Glucuronate isomerase EC: 5.3.1.12
Arabinose-5-phosphate isomerase EC: 5.3.1.13
L-rhamnose isomerase EC: 5.3.1.14
D-lyxose ketol-isomerase EC: 5.3.1.15
1-(5-phosphoribosyl)-5-((5-phosphoribosylamino)methylideneamino)imidazole-4-carboxamideisomerase EC
5-dehydro-4-deoxy-D-glucuronate isomerase EC: 5.3.1.17

Selected 1 item

Filter: All

Glucose-6-phosphate isomerase
EC: 5.3.1.9 Brenda Enzymes

Alternative names
Hexose monophosphate isomerase
Hexosephosphate isomerase
Oxoisomerase
Phosphoglucoisomerase
Phosphoglucose isomerase
Phosphohexoisomerase
Phosphohexomutase
Phosphohexose isomerase
Phosphosaccharomutase

Catalytic activity
D-glucose 6-phosphate = D-fructose 6-phosphate

1 element(s) are selected

Figure 16.4: *The EC Database View*

The **Search** field can be used to search in EC term names and descriptions. The **Previous** and **Next** buttons can be used to navigate to the matching EC terms.

When multiple items are selected, the arrow buttons at the bottom of the view can be used to focus on the different selected elements - note, that this does not change the selection, it only changes the focused area of the tree.

It is possible to **Select Names in Other Views** and **Copy Names to Clipboard** (see figure 16.4). Selecting names in other views will match names in other editors that support this - currently, this is only supported in the Differential Abundance element view.

The **Filter** side panel section can be used to restrict the view to the terms that are currently selected. This is in particular useful, when EC terms from an another editor (such as the Differential Abundance element view) have been selected.

When clicking on a EC term, the Property viewer in the side panel will show a description together with related links to more information.

16.3 Download Pathway Database

The Download Pathway Database tool allows you to download pathway databases for use with the Identify Pathways tool.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Functional Analysis** ( ) | **Download Pathway Database** ( )

In the first wizard step, select the pathway database to download. Currently only one database is available:

- MetaCyc Pathway Database: A multi-organism database of pathways involved in primary and secondary metabolism, as well as associated metabolites, reactions, enzymes, and genes. The MetaCyc database consists of pathways included in the MetaCyc BioPAX Level 3 file (https://metacyc.org/).

16.3.1 The Pathway Database

A pathway database is a clc object that contains information about pathways, specifically which functional terms are present in each pathway, e.g. for MetaCyc each pathway is associated with a set of EC numbers. These EC numbers form the basis for pathway calling from an abundance table or a differential abundance table.

The pathway database has three views:

- The Pathway table ( )
- The Compound table ( )
- The Enzyme table ( )

The Pathway table The pathway table is a list of all pathways in the database and it has two columns, Name and MetaCyc ID, giving the name of a pathway and its MetaCyc ID, which is also a link to the relevant online site. The pathway table has two buttons at the bottom

- **Open Pathway View**: Opens a split view with a visualization of the selected pathways. Note that double clicking on a single line in the Pathway table is identical to selecting that line and clicking on Open Pathway View.
- **Create New Pathway Database**: Creates a new database only containing the selected pathways. This is useful, e.g. for constructing a species specific database

The Compound table The compound table is a list of all compounds, including side compounds, in the database. This table has three columns,

- **Name**: The name of the compound.
- **MetaCyc ID**: The compound's MetaCyc ID which is a link to the compound's description on the MetaCyc homepage.
- **Cellular location**: The location of the compound in the cell. Some names occur several times as their cellular locations are different, e.g. there are protons in the cytosol, outside of the cell and inside the lumen of some organelles.

Furthermore, the table has two buttons in the bottom

- **Open Pathway View from Selected Compound(s)**: Opens a split view with a visualization of the pathways in which the selected compounds occur. Note that double clicking on a single line in the Compund table is identical to selecting that line and clicking on Open Pathway View from Selected Compound(s).
- **Create New Pathway Database**: Creates a new pathway database containing only pathways in which the selected compounds occur.

The Enzyme table The enzyme table is a list of all enzymes in the database. This table has three columns,

- **Name**: the name of the Enzyme. Note that some names occur several times because they reference different genes.
- **MetaCyc ID**: the enzyme's MetaCyc ID which is a link to the enzyme's description on the MetaCyc homepage.
- **Cellular location**: the location of the enzyme in the cell. Some names occur several times as their cellular locations are different, e.g. there are protons in the cytosol, outside of the cell and inside the lumen of some organelles.

Furthermore, the table has two buttons in the bottom

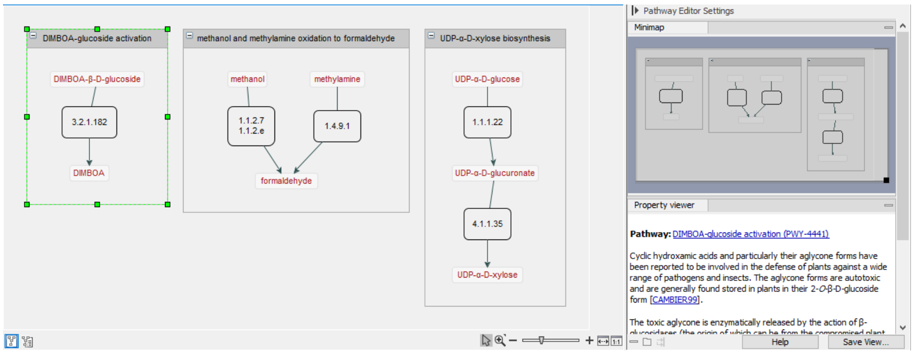
- **Open Pathway View from Selected Enzyme(s)**: Opens a split view with a visualization of the pathways in which the selected enzymes occur. Note that double clicking on a single line in the Enzyme table is identical to selecting that line and clicking on Open Pathway View from Selected Enzyme(s).
- **Create New Pathway Database**: Creates a new pathway database containing only pathways in which the selected enzymes occur.

16.3.2 The Pathway View

When opening one or several pathways in Pathway View, the pathways may be explored visually. Note that the pathway view has two editors

- The Pathway Graph View ( )
- The Text Contents View ( )

The Pathway Graph View The pathway graph view graphically shows a biochemical pathway in the form of a simplistic textbook style pathway drawing, where each reaction is labeled by one or several EC numbers in black rounded boxes, and reaction arrows connect the reactants, products and EC numbers. As reactants, only the main compounds are shown, side-compounds are ignored in the visualization. All reactants are visible in the property view when selecting an EC number in the pathway view. This information is also displayed as a tool tip when hovering over the EC number.



Pathway: DIMBOA-glucoside activation (PWY-4441)

Cyclic hydroxamic acids and particularly their aglycone forms have been reported to be involved in the defense of plants against a wide range of pathogens and insects. The aglycone forms are autotoxic and are generally found stored in plants in their 2-*O*-β-D-glucoside form [CAMBIER99].

The toxic aglycone is enzymatically released by the action of β-glucosidases (the origin of which can be from the compromised plant

Figure 16.5: *The Pathway view*

Upon clicking an item in the pathway graph additional information about that element will be displayed in the Property Viewer of the side panel.

The Text Contents The text contents view of a pathway contains a textual summary of the pathways including scientific references.

16.4 Create DIAMOND Index

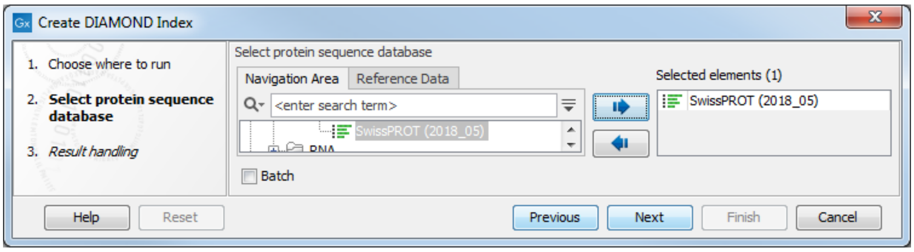
This tool will compute a DIAMOND index from a protein database. These indexes can then be used as input to the Annotate CDS with Best DIAMOND Hit tool (see section 11.5) and Annotate with DIAMOND tool (see section 11.3).

If the input sequences contain metadata, such as GO-terms, these will be transferred to the created index.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Functional Analysis** ( ) | **Create DIAMOND Index** ( )

In the first dialog, select a protein database downloaded with the Download Protein Database tool (figure 16.6).



Create DIAMOND Index

1. Choose where to run
2. **Select protein sequence database**
3. *Result handling*

Select protein sequence database

Navigation Area | Reference Data

<enter search term>

SwissPROT (2018_05)

RNA

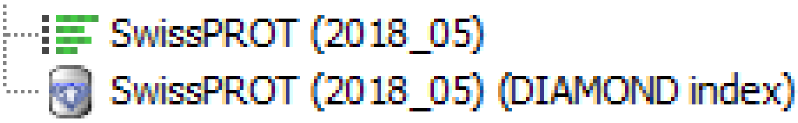
Selected elements (1)

SwissPROT (2018_05)

□ Batch

Figure 16.6: *Select a protein database.*

The output is an index file as seen in figure 16.7.

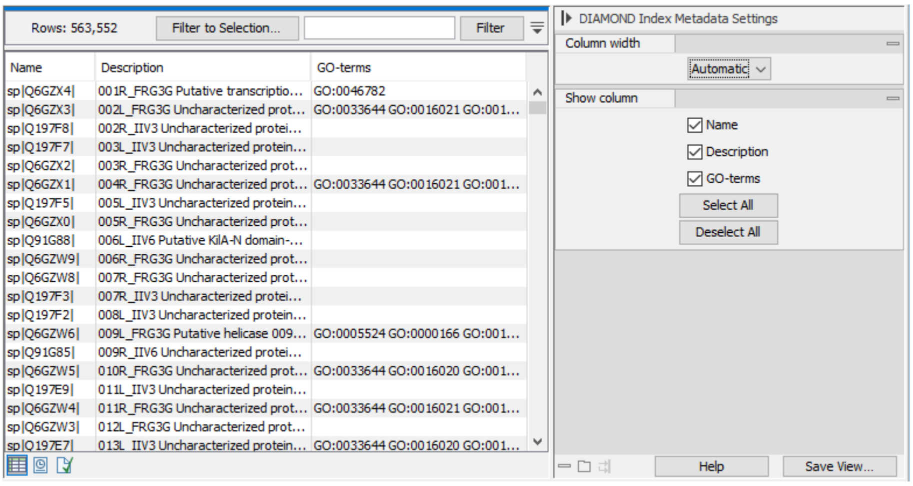


SwissPROT (2018_05)

SwissPROT (2018_05) (DIAMOND index)

Figure 16.7: *The protein database and its index as seen in the Navigation Area.*

The default view for the DIAMOND index is a tabular overview of the sequence entries and their associated metadata, such as GO-terms (see figure 16.8).



Rows: 563,552

Name	Description	GO-terms
sp|Q6GZX4|	001R_FRG3G Putative transcriptio...	GO:0046782
sp|Q6GZX3|	002L_FRG3G Uncharacterized prot...	GO:0033644 GO:0016021 GO:001...
sp|Q197F8|	002R_IIV3 Uncharacterized protei...	
sp|Q197F7|	003L_IIV3 Uncharacterized protein...	
sp|Q6GZX2|	003R_FRG3G Uncharacterized prot...	
sp|Q6GZX1|	004R_FRG3G Uncharacterized prot...	GO:0033644 GO:0016021 GO:001...
sp|Q197F5|	005L_IIV3 Uncharacterized protein...	
sp|Q6GZX0|	005R_FRG3G Uncharacterized prot...	
sp|Q91G88|	006L_IIV6 Putative KilA-N domain-...	
sp|Q6GZW9|	006R_FRG3G Uncharacterized prot...	
sp|Q6GZW8|	007R_FRG3G Uncharacterized prot...	
sp|Q197F3|	007R_IIV3 Uncharacterized protei...	
sp|Q197F2|	008L_IIV3 Uncharacterized protein...	
sp|Q6GZW6|	009L_FRG3G Putative helicase 009...	GO:0005524 GO:0000166 GO:001...
sp|Q91G85|	009R_IIV6 Uncharacterized protei...	
sp|Q6GZW5|	010R_FRG3G Uncharacterized prot...	GO:0033644 GO:0016020 GO:001...
sp|Q197E9|	011L_IIV3 Uncharacterized protein...	
sp|Q6GZW4|	011R_FRG3G Uncharacterized prot...	GO:0033644 GO:0016021 GO:001...
sp|Q6GZW3|	012L_FRG3G Uncharacterized prot...	
sp|Q197E7|	013L_IIV3 Uncharacterized protein...	GO:0033644 GO:0016020 GO:001...

DIAMOND Index Metadata Settings

Column width: Automatic

Show column

☑ Name
☑ Description
☑ GO-terms

Figure 16.8: *The default view for the DIAMOND index element.*

16.5 Import RNAcentral Database

This tool can be used to import non-coding RNA sequences from RNAcentral, and join the sequences with functional Gene Ontology information.

The imported sequences can then be used together with the Annotate with BLAST tool (see section 11.2) and the Build Functional Profile tool (see section 11.7) to quantify the functional annotation abundances.

The Import RNAcentral Database tool uses a special FASTA importer that allows for non-standard nucleotides (RNAcentral includes sequences with non-standard IUPAC nucleotide symbols, which are not allowed by our standard FASTA importer).

The tool can also import RNAcentral files with associations to GO-terms, such as 'rnacentral_rfam_annotations.tsv.gz', and match the entries with those in the imported sequence list.

Before running the tool, it is necessary to download the relevant sequences and GO-associations from RNAcentral (`https://rnacentral.org/`). To get the full set of annotations, we recommend downloading the following files:

RNAcentral FASTA sequences: `ftp://ftp.ebi.ac.uk/pub/databases/RNAcentral/current_release/sequences/rnacentral_active.fasta.gz`

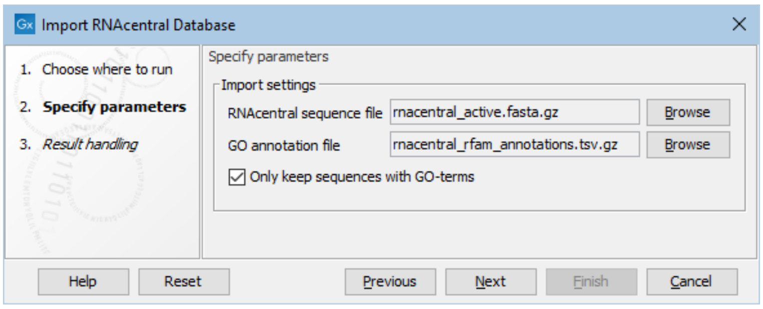
RNAcentral GO Associations (from RFAM): `ftp://ftp.ebi.ac.uk/pub/databases/RNAcentral/current_release/go_annotations/rnacentral_rfam_annotations.tsv.gz`

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Functional Analysis** ( ) | **Import RNAcentral Database** ( )

In the tool dialog (figure 16.9), select the files downloaded as described above.

It is also possible to select whether to include only RNAcentral sequences with matching GO associations, which will reduce the size of the created database.


Import RNAcentral Database

1. Choose where to run
2. **Specify parameters**
3. *Result handling*

Specify parameters

Import settings

RNAcentral sequence file: rnacentral_active.fasta.gz

GO annotation file: rnacentral_rfam_annotations.tsv.gz

☑ Only keep sequences with GO-terms

Figure 16.9: *The Import RNAcentral Database tool options.*

RNAcentral identifiers may contain a species-specific suffix (e.g. URS0000000006_1317357 - here 1317357 is an NCBI Taxonomy ID). When we perform the matching of RNAcentral sequences to GO associations these are stripped off and ignored.

16.6 Import PICRUSt2 Multiplication Table

The Import PICRUSt2 Multiplication Table (beta) tool can be used to import multiplication tables from PICRUSt2 [Douglas et al., 2020] in order to perform functional inference for OTU abundance tables using Infer Functional Profile (beta) 11.8 or to normalize OTU abundance tables by rRNA copy numbers using Normalize OTU Table by Copy Number (beta) 16.6.

Before running the tool it is necessary to download the relevant data files from the PICRUSt2 github repository (`https://github.com/picrust/picrust2/tree/master/picrust2/default_files`), specifically three kinds of files are required:

1. Files with 16S, 18S or ITS sequence alignments
    - 16S alignments or prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/pro_ref/pro_ref.fna)
    - 18S alignments for fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/fungi_18S/fungi_18S.fna.gz)
    - ITS alignments for fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/fungi_ITS/fungi_ITS.fna.gz)
2. Files with rRNA copy numbers
    - 16S rRNA copy numbers for prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/16S.txt.gz)
    - 18S rRNA copy numbers for fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/18S_counts.txt.gz)
    - ITS rRNA copy numbers for fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/ITS_counts.txt.gz)
3. Files with functional term counts associated with each type of rRNA
    - EC terms associated with 16S regions in prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/ec.txt.gz)
    - Kegg orthology terms associated with 16S regions in prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/ko.txt.gz)
    - COG terms associated with 16S regions in prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/cog.txt.gz)
    - Pfam domains associated with 16S regions in prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/pfam.txt.gz)
    - TIGRFAM terms associated with 16S regions in prokaryotes (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/prokaryotic/tigrfam.txt.gz)
    - EC terms associated with 18S regions in fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/ec_18S_counts.txt.gz)
    - EC terms associated with ITS regions in fungi (https://github.com/picrust/picrust2/blob/master/picrust2/default_files/fungi/ec_ITS_counts.txt.gz)

Only files corresponding to the same rRNA regions can be combined to obtain a valid PICRUSt2 Multiplication Table, e.g. 16S alignments for prokaryotes, 16S rRNA counts and COG terms associated with 16S regions in prokaryotes.

Note that the rRNA copy numbers for fungi 2 are not consistent for 18S and ITS regions, which may have implications for the normalization and thus also for the functional inference for fungal data.

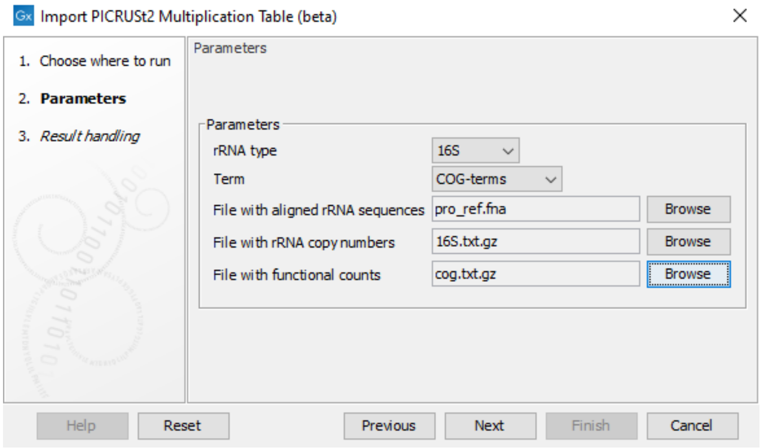
The tool can import similarly prepared data if other data sources are available. The OTU sequences need not be aligned.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Functional Analysis** ( ) | **Import PICRUSt2 Multiplication Table (beta)** ( )

In the tool dialog (figure 16.10), select which type of rRNA and which type of terms you would like to import, then select the corresponding three files downloaded above where

- File with aligned rRNA sequences: takes fasta files as input, e.g. one of the files listed under point 1.
- File with rRNA copy numbers: takes a tab separated text file with two columns as input, where the first column contains the name of an rRNA sequence from the fasta file and the second column the corresponding rRNA copy number, e.g. one of the files listed under point 2 or a fasta file with unaligned rRNA sequences. The file is expected to contain a header.
- File with functional counts: takes a tab separated text file as input. The first column contains the name of an rRNA sequence from the fasta file and the remaining columns contain the corresponding functional counts, where each column is identified with a functional term via the header line, e.g. one of the files listed under point 3.



Parameters

rRNA type: 16S

Term: COG-terms

File with aligned rRNA sequences: pro_ref.fna

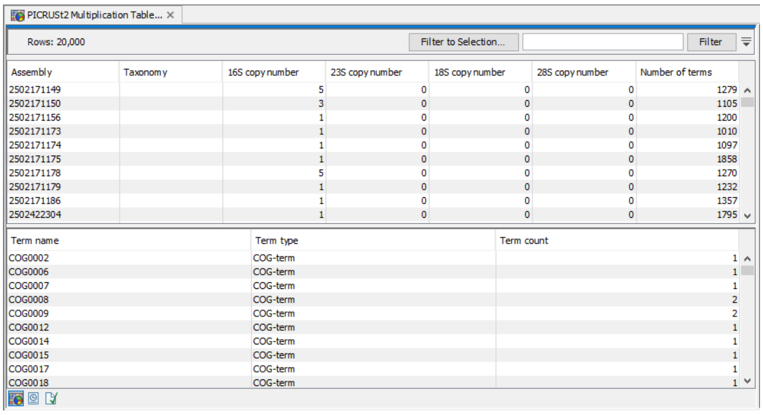
File with rRNA copy numbers: 16S.txt.gz

File with functional counts: cog.txt.gz

Figure 16.10: *The Import PICRUSt2 Multiplication Table (beta) tool options.*

After import the data in the multiplication table is displayed in a table (figure 16.11) where the name of the rRNA is given under the "Assembly" column, as these tables are typically derived from assemblies with known rRNA content, taxonomy and functional counts. The following four columns list the rRNA copy numbers registered for each type of rRNA, ITS regions will be listed as the selected rRNA type, e.g. 18S or 28S and the number of distinct funtional terms registered for that assembly in the column "Number of terms".

When selecting one or several rows in the upper table, the lower table will show the combined functional counts for the selected row for each of the functional terms individually.



Rows: 20,000

Assembly	Taxonomy	16S copy number	23S copy number	18S copy number	28S copy number	Number of terms
2502171149		5	0	0	0	1279
2502171150		3	0	0	0	1105
2502171156		1	0	0	0	1200
2502171173		1	0	0	0	1010
2502171174		1	0	0	0	1097
2502171175		1	0	0	0	1858
2502171178		5	0	0	0	1270
2502171179		1	0	0	0	1232
2502171186		1	0	0	0	1357
2502422304		1	0	0	0	1795

Term name	Term type	Term count
COG0002	COG-term	1
COG0006	COG-term	1
COG0007	COG-term	1
COG0008	COG-term	2
COG0009	COG-term	2
COG0012	COG-term	1
COG0014	COG-term	1
COG0015	COG-term	1
COG0017	COG-term	1
COG0018	COG-term	1

Figure 16.11: *The PICRUSt2 Multiplication Table visualization.*

Chapter 17

Databases for Drug Resistance Analysis

17.1 Download Resistance Database

Download Resistance Database enables download of databases for use with the Find Resistance with Nucleotide Database, Find Resistance with PointFinder and Find Resistance with ShortBRED tools.

To run the tool, go to:

> **Toolbox** | **Microbial Genomics Module** ( ) | **Databases** ( ) | **Drug Resistance Analysis** ( ) | **Download Resistance Database** ( )

The available databases fall into four categories:

ShortBRED Marker Databases

These databases can be used with **Find Resistance with ShortBRED** (section 12.3). The databases are marker databases, containing peptide fragments that uniquely characterize sets of similar proteins, rather than a specific gene.

- **QMI-AR Peptide Marker Database.** The QIAGEN Microbial Insight - Antimicrobial Resistance database is a curated database containing peptide markers derived from the following source databases: CARD (https://card.mcmaster.ca/), ARG-ANNOT [Gupta et al., 2014] (https://www.mediterranee-infection.com/acces-ressources/base-de-donnees/arg-annot-2/), NCBI Bacterial Antimicrobial Resistance Reference Gene Database (https://www.ncbi.nlm.nih.gov/bioproject/PRJNA313047), ResFinder (https://bitbucket.org/genomicepidemiology/resfinder_db/src/master/).
- **CARD Peptide Marker Database.** Peptide markers derived from the Comprehensive Antibiotic Resistance Database (CARD) (https://card.mcmaster.ca/).
- **ARG-ANNOT Peptide Marker Database.** Protein markers from the ARG-ANNOT database [Gupta et al., 2014] (https://www.mediterranee-infection.com/acces-ressources/base-de-donnees/arg-annot-2/).

Nucleotide Databases

These databases can be used with **Find Resistance with Nucleotide Database** (section 12.2). The databases contain nucleotide gene sequences.

- **QMI-AR Nucleotide Database**. The QIAGEN Microbial Insight - Antimicrobial Resistance database is a curated database containing nucleotide sequences compiled from the following source databases: CARD (https://card.mcmaster.ca/), ARG-ANNOT [Gupta et al., 2014] (https://www.mediterranee-infection.com/acces-ressources/base-de-donnees/arg-annot-2/), NCBI Bacterial Antimicrobial Resistance Reference Gene Database (https://www.ncbi.nlm.nih.gov/bioproject/PRJNA313047), ResFinder (https://bitbucket.org/genomicepidemiology/resfinder_db/src/master/).
- **VFDB**. Nucleotide sequences compiled from the Virulence Factor database core dataset (https://www.mgc.ac.cn/VFs/download.htm).
- **CARD Nucleotide Database**. Nucleotide sequences compiled from the Comprehensive Antibiotic Resistance Database (CARD) (https://card.mcmaster.ca/).
- **ResFinder Nucleotide Database**. Nucleotide sequences from the ResFinder database (https://bitbucket.org/genomicepidemiology/resfinder_db/src/master/).

Point Mutation Databases

These databases are used with **Find Resistance with PointFinder** (section 12.1).

- **PointFinder**. Organism-specific databases containing wild type genes and known resistance conferring mutations [Zankari et al., 2017] (https://bitbucket.org/genomicepidemiology/).

Integrated Databases

- **ARESdb**. The ARES Database is an interactive interface to ARESdb (https://www.opgen.com/ares/ares-products/aresdb/), a curated database of computationally inferred antimicrobial resistance genes and their respective predictive performances, both established and provided by Ares Genetics. With the ARES Database it is possible to produce sequence lists which are compatible with the Find Resistance with Nucleotide Database and Find Resistance with PointFinder tools, but also with other tools taking annotated sequence lists as input.

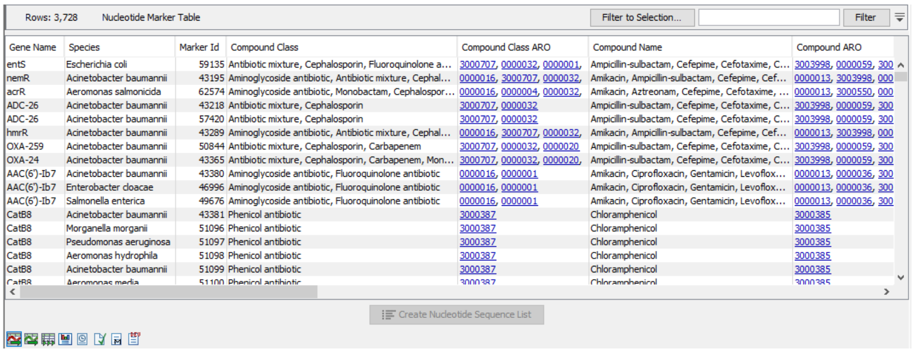
17.1.1 ARES Database

The ARES Database has three table views:

- a Nucleotide Marker table for gene markers. From this view, it is possible to extract a sequence list which may be used with the Find Resistance with Nucleotide Database tool.
- a Protein Marker table for gene markers.
- a Point Mutation Marker table for Single Nucleotide Polymorphism (SNP) markers. From this view, it is possible to extract a sequence list which may be used with the Find Resistance with PointFinder tool.

The database also comes with an overview report summarizing its content.

The Nucleotide Marker table The Nucleotide Marker Table ( ) lists AMR marker genes together with AMR related information annotated with CARD Antibiotic Resistance Ontology (ARO) accession numbers (figure 17.1).



Rows: 3,728 Nucleotide Marker Table

Gene Name	Species	Marker Id	Compound Class	Compound Class ARO	Compound Name	Compound ARO
entS	Escherichia coli	59135	Antibiotic mixture, Cephalosporin, Fluoroquinolone a...	3000707, 0000032, 0000001,	Ampicillin-sulbactam, Cefepime, Cefotaxime, C...	3003998, 0000059, 300
nemR	Acinetobacter baumannii	43195	Aminoglycoside antibiotic, Antibiotic mixture, Cephal...	0000016, 3000707, 0000032,	Amikacin, Ampicillin-sulbactam, Cefepime, Cef...	0000013, 3003998, 000
acrR	Aeromonas salmonicida	62574	Aminoglycoside antibiotic, Monobactam, Cephalospor...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxime, ...	0000013, 3000550, 000
ADC-26	Acinetobacter baumannii	43218	Antibiotic mixture, Cephalosporin	3000707, 0000032	Ampicillin-sulbactam, Cefepime, Cefotaxime, C...	3003998, 0000059, 300
ADC-26	Acinetobacter baumannii	57420	Antibiotic mixture, Cephalosporin	3000707, 0000032	Ampicillin-sulbactam, Cefepime, Cefotaxime, C...	3003998, 0000059, 300
hmrR	Acinetobacter baumannii	43289	Aminoglycoside antibiotic, Antibiotic mixture, Cephal...	0000016, 3000707, 0000032,	Amikacin, Ampicillin-sulbactam, Cefepime, Cef...	0000013, 3003998, 000
OXA-259	Acinetobacter baumannii	50844	Antibiotic mixture, Cephalosporin, Carbapenem	3000707, 0000032, 0000020	Ampicillin-sulbactam, Cefepime, Cefotaxime, C...	3003998, 0000059, 300
OXA-24	Acinetobacter baumannii	43365	Antibiotic mixture, Cephalosporin, Carbapenem, Mon...	3000707, 0000032, 0000020,	Ampicillin-sulbactam, Cefepime, Cefotaxime, C...	3003998, 0000059, 300
AAC(6')-Ib7	Acinetobacter baumannii	43380	Aminoglycoside antibiotic, Fluoroquinolone antibiotic	0000016, 0000001	Amikacin, Ciprofloxacin, Gentamicin, Levoflox...	0000013, 0000036, 300
AAC(6')-Ib7	Enterobacter cloacae	46996	Aminoglycoside antibiotic, Fluoroquinolone antibiotic	0000016, 0000001	Amikacin, Ciprofloxacin, Gentamicin, Levoflox...	0000013, 0000036, 300
AAC(6')-Ib7	Salmonella enterica	49676	Aminoglycoside antibiotic, Fluoroquinolone antibiotic	0000016, 0000001	Amikacin, Ciprofloxacin, Gentamicin, Levoflox...	0000013, 0000036, 300
CatB8	Acinetobacter baumannii	43381	Phenicol antibiotic	3000387	Chloramphenicol	3000385
CatB8	Morganella morganii	51096	Phenicol antibiotic	3000387	Chloramphenicol	3000385
CatB8	Pseudomonas aeruginosa	51097	Phenicol antibiotic	3000387	Chloramphenicol	3000385
CatB8	Aeromonas hydrophila	51098	Phenicol antibiotic	3000387	Chloramphenicol	3000385
CatB8	Acinetobacter baumannii	51099	Phenicol antibiotic	3000387	Chloramphenicol	3000385
CatB8	Aeromonas media	51100	Phenicol antibiotic	3000387	Chloramphenicol	3000385

Figure 17.1: *A Nucleotide Marker Table.*

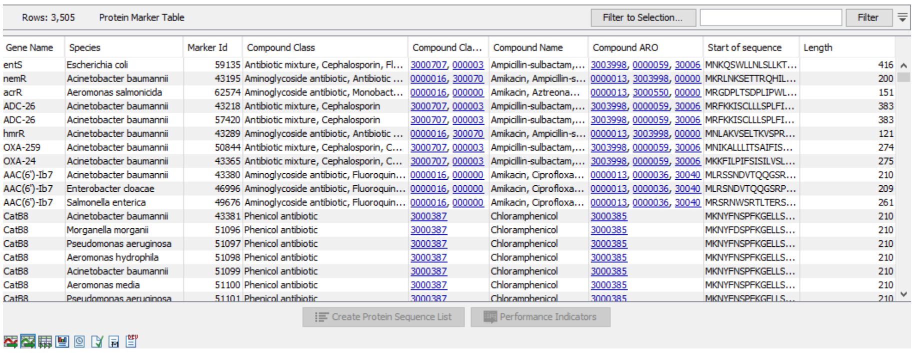
The table contains the following columns:

- **Gene Name**. The name of the resistance gene or a generic unique name starting with "group_" if no relevant gene information is available.
- **Species**. The name of the species for which a given resistance gene was found. The presence of a gene marker in a species different from the one listed in this column may still indicate AMR, see performance table below.
- **Marker Id**. An identification number for the peptide sequence of the nucleotide sequence.
- **Compound Class**. A list of compound classes, corresponding to the compounds in Compound Name.
- **Compound Class ARO**. The ARO accession numbers corresponding to the Compound Class.
- **Compound Name**. A list of antibiotic compounds to which the gene marker confers resistance.
- **Compound ARO**. The ARO accession numbers corresponding to the Compound Name.
- **Start of sequence**. The beginning of the gene marker sequence.
- **Length**. The length of the gene marker in number of nucleotides.

Click on **Create a Nucleotide Sequence List** to create a sequence list with unique sequences that can be used as a database for the Find Resistance with Nucleotide Database tool. Note that the underlying sequences can be the same for two different species.

The Protein Marker table The Protein Marker Table ( ) lists the protein products corresponding to the sequences of the Nucleotide Marker Table (figure 17.2). The Marker Id shows the connection between nucleotide and protein markers. Note that selections in the nucleotide and protein tables are synchronized when both views are open simultaneously.

The table contains the following information:



Rows: 3,505 Protein Marker Table

Gene Name	Species	Marker Id	Compound Class	Compound Cla...	Compound Name	Compound ARO	Start of sequence	Length
entS	Escherichia coli	59135	Antibiotic mixture, Cephalosporin, Fl...	3000707, 000003	Ampicillin-sulbactam,...	3003998, 0000059, 30006	MNKQSWLLNLSLLKT...	416
nemR	Acinetobacter baumannii	43195	Aminoglycoside antibiotic, Antibiotic ...	0000016, 300070	Amikacin, Ampicillin-s...	0000013, 3003998, 00000	MKRLNKSETTRQHIL...	200
acrR	Aeromonas salmonicida	62574	Aminoglycoside antibiotic, Monobact...	0000016, 000000	Amikacin, Aztreona...	0000013, 3000550, 00000	MRGDPLTSDPLIPWL...	151
ADC-26	Acinetobacter baumannii	43218	Antibiotic mixture, Cephalosporin	3000707, 000003	Ampicillin-sulbactam,...	3003998, 0000059, 30006	MRFKKISCLLLSPLFI...	383
ADC-26	Acinetobacter baumannii	57420	Antibiotic mixture, Cephalosporin	3000707, 000003	Ampicillin-sulbactam,...	3003998, 0000059, 30006	MRFKKISCLLLSPLFI...	383
hmrR	Acinetobacter baumannii	43289	Aminoglycoside antibiotic, Antibiotic ...	0000016, 300070	Amikacin, Ampicillin-s...	0000013, 3003998, 00000	MNLAKVSELTKVSPR...	121
OXA-259	Acinetobacter baumannii	50844	Antibiotic mixture, Cephalosporin, C...	3000707, 000003	Ampicillin-sulbactam,...	3003998, 0000059, 30006	MNIKALLLITSAIFIS...	274
OXA-24	Acinetobacter baumannii	43365	Antibiotic mixture, Cephalosporin, C...	3000707, 000003	Ampicillin-sulbactam,...	3003998, 0000059, 30006	MKKFILPIFSISILVSL...	275
AAC(6')-Ib7	Acinetobacter baumannii	43380	Aminoglycoside antibiotic, Fluoroquin...	0000016, 000000	Amikacin, Ciprofloxa...	0000013, 0000036, 30040	MLRSSNDVTQQGSR...	210
AAC(6')-Ib7	Enterobacter cloacae	46996	Aminoglycoside antibiotic, Fluoroquin...	0000016, 000000	Amikacin, Ciprofloxa...	0000013, 0000036, 30040	MLRSNDVTQQGSRP...	209
AAC(6')-Ib7	Salmonella enterica	49676	Aminoglycoside antibiotic, Fluoroquin...	0000016, 000000	Amikacin, Ciprofloxa...	0000013, 0000036, 30040	MRSRNWSRTLTERS...	261
CatB8	Acinetobacter baumannii	43381	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210
CatB8	Morganella morganii	51096	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFDSPFKGELLS...	210
CatB8	Pseudomonas aeruginosa	51097	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210
CatB8	Aeromonas hydrophila	51098	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210
CatB8	Acinetobacter baumannii	51099	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210
CatB8	Aeromonas media	51100	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210
CatB8	Pseudomonas aeruginosa	51101	Phenicol antibiotic	3000387	Chloramphenicol	3000385	MKNYFNSPFKGELLS...	210

Figure 17.2: *A Protein Marker Table.*

- **Gene Name**. The name of the resistance gene or a generic unique name starting with "group_" if no relevant gene information is available. Note that this does not necessarily correspond to the "group_" name of the nucleotide sequence as the same Marker Id can occur for multiple nucleotide sequences.

- **Species**. A list of species for which a given resistance gene was found in the database. The presence of a gene marker in a species different from the ones listed in this column may still indicate AMR, see the Performance Table View section below.

- **Marker Id**. An identification number for the peptide sequence of the nucleotide sequence.

- **Compound Class**. A list of compound classes, corresponding to the compounds in Compound Name.

- **Compound Class ARO**. The ARO accession numbers corresponding to the Compound Class.

- **Compound Name**. A list of antibiotic compounds to which the gene marker confers resistance.

- **Compound ARO**. The ARO accession numbers corresponding to the Compound Name.

- **Start of sequence**. The beginning of the gene marker sequence.

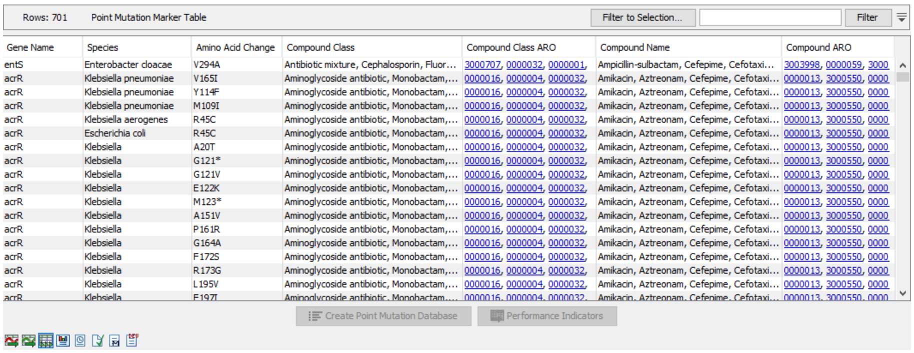
- **Length**. The length of the gene marker in number of nucleotides.

From this table, it is possible to **Create a Protein Sequence List**, i.e., an annotated protein sequence list of gene markers for the proteins and species of interest. Furthermore, the button **Performance Indicators** opens another table with the performance statistics obtained from experimental data.

The Point Mutation Marker table The Point Mutation Marker table (▦) gives an overview of resistance conferring single-nucleotide polymorphisms (SNP's) and their performance data (figure 17.3).

The columns of the Point Mutation Marker Table are:

- **Gene Name**. The name of the resistance gene or a generic unique name starting with "group_" if no relevant gene information is available.



Gene Name	Species	Amino Acid Change	Compound Class	Compound Class ARO	Compound Name	Compound ARO
entS	Enterobacter cloacae	V294A	Antibiotic mixture, Cephalosporin, Fluor...	3000707, 0000032, 0000001,	Ampicillin-sulbactam, Cefepime, Cefotaxi...	3003998, 0000059, 3000
acrR	Klebsiella pneumoniae	V165I	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella pneumoniae	Y114F	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella pneumoniae	M109I	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella aerogenes	R45C	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Escherichia coli	R45C	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	A20T	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	G121*	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	G121V	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	E122K	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	M123*	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	A151V	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	P161R	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	G164A	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	F172S	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	R173G	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	L195V	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000
acrR	Klebsiella	F197I	Aminoglycoside antibiotic, Monobactam,...	0000016, 0000004, 0000032,	Amikacin, Aztreonam, Cefepime, Cefotaxi...	0000013, 3000550, 0000

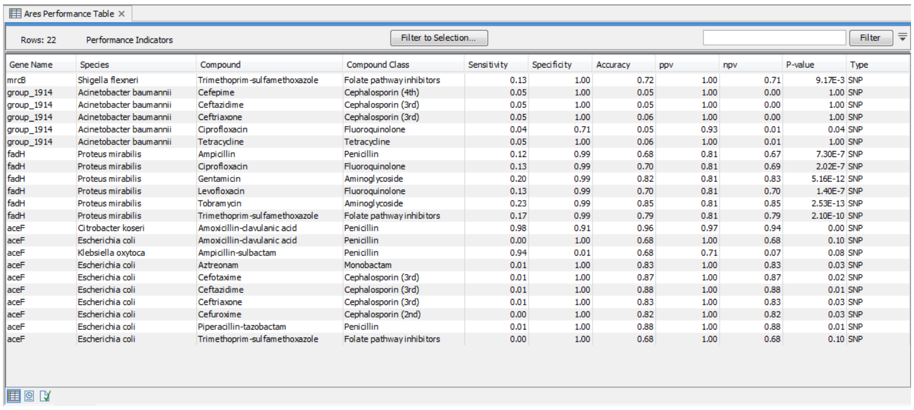
Figure 17.3: *A Point Mutation Marker Table.*

- **Species**. A list of species for which a given resistance gene was found in the database. The presence of a point mutation in a species different from the ones listed in this column may still indicate AMR, see the Performance Table View section below.
- **Amino Acid Change**. This describes the change on the protein level.
- **Compound Class**. A list of compound classes, corresponding to the compounds in Compound Name.
- **Compound Class ARO**. The ARO accession numbers corresponding to the Compound Class.
- **Compound Name**. A list of antibiotic compounds to which the point mutation confers resistance.
- **Compound ARO**. The ARO accession numbers corresponding to the Compound Name.

From this table it is possible to create a sequence list which can be used as a database for the Find Resistance with PointFinder tool by pressing the **Create Point Mutation Database** button and each mutation is associated with performance data which can be accessed by clicking the button **Performance Indicators**.

The Performance Indicators table Microbial susceptibility to various compounds may depend on a combination of multiple genetic factors and/or be specific to certain organism. As such, the presence of a single gene or variant marker is not a clear indication of whether an isolate is resistant to compound or class or class of compounds. The ARES Performance Indicators table (figure 17.4) provides a measure for how well a given gene marker is expected to perform for a given compound and species. The gene marker statistics is not only calculated for the species the marker was originally identified in, but for other species as well.

The ARES Performance Indicators tables were derived by identifying the gene markers in a large collection of isolates with known resistance profiles. Based on the presence of the gene markers in the isolates, confusion matrix statistics (https://en.wikipedia.org/wiki/Confusion_matrix) were calculated for a large number of marker, species, and compound combinations. In confusion matrix terms, resistant isolates are considered 'positives' and susceptible isolates are considered 'negatives'. For example, a resistant isolate with a given gene marker is considered a 'true positive', and a resistant isolate without a given gene marker is a 'false negative'.



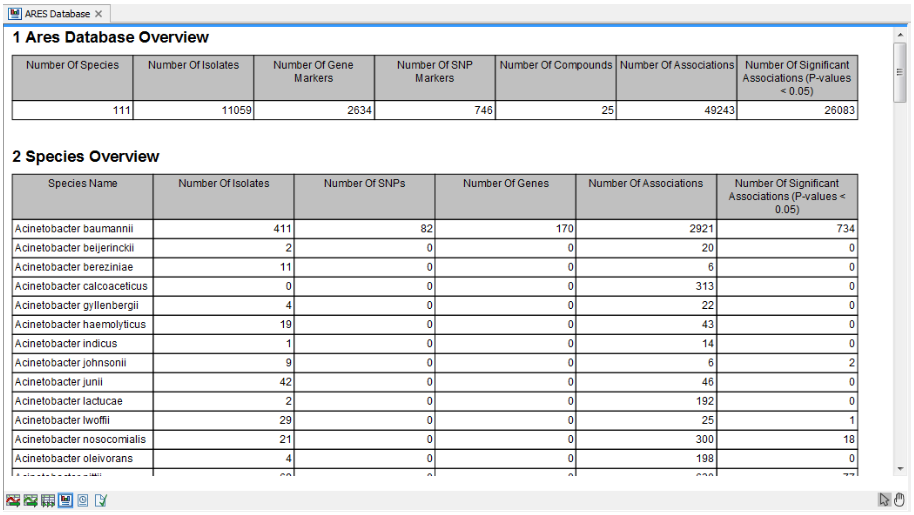
Gene Name	Species	Compound	Compound Class	Sensitivity	Specificity	Accuracy	ppv	npv	P-value	Type
mrcB	Shigella flexneri	Trimethoprim-sulfamethoxazole	Folate pathway inhibitors	0.13	1.00	0.72	1.00	0.71	9.17E-3	SNP
group_1914	Acinetobacter baumannii	Cefepime	Cephalosporin (4th)	0.05	1.00	0.05	1.00	0.00	1.00	SNP
group_1914	Acinetobacter baumannii	Ceftazidime	Cephalosporin (3rd)	0.05	1.00	0.05	1.00	0.00	1.00	SNP
group_1914	Acinetobacter baumannii	Ceftriaxone	Cephalosporin (3rd)	0.05	1.00	0.06	1.00	0.00	1.00	SNP
group_1914	Acinetobacter baumannii	Ciprofloxacin	Fluoroquinolone	0.04	0.71	0.05	0.93	0.01	0.04	SNP
group_1914	Acinetobacter baumannii	Tetracycline	Tetracycline	0.05	1.00	0.06	1.00	0.01	1.00	SNP
fadH	Proteus mirabilis	Ampicillin	Penicillin	0.12	0.99	0.68	0.81	0.67	7.30E-7	SNP
fadH	Proteus mirabilis	Ciprofloxacin	Fluoroquinolone	0.13	0.99	0.70	0.81	0.69	2.02E-7	SNP
fadH	Proteus mirabilis	Gentamicin	Aminoglycoside	0.20	0.99	0.82	0.81	0.83	5.16E-12	SNP
fadH	Proteus mirabilis	Levofloxacin	Fluoroquinolone	0.13	0.99	0.70	0.81	0.70	1.40E-7	SNP
fadH	Proteus mirabilis	Tobramycin	Aminoglycoside	0.23	0.99	0.85	0.81	0.85	2.53E-13	SNP
fadH	Proteus mirabilis	Trimethoprim-sulfamethoxazole	Folate pathway inhibitors	0.17	0.99	0.79	0.81	0.79	2.10E-10	SNP
aceF	Citrobacter koseri	Amoxicillin-clavulanic acid	Penicillin	0.98	0.91	0.96	0.97	0.94	0.00	SNP
aceF	Escherichia coli	Amoxicillin-clavulanic acid	Penicillin	0.00	1.00	0.68	1.00	0.68	0.10	SNP
aceF	Klebsiella oxytoca	Ampicillin-sulbactam	Penicillin	0.94	0.01	0.68	0.71	0.07	0.08	SNP
aceF	Escherichia coli	Aztreonam	Monobactam	0.01	1.00	0.83	1.00	0.83	0.03	SNP
aceF	Escherichia coli	Cefotaxime	Cephalosporin (3rd)	0.01	1.00	0.87	1.00	0.87	0.02	SNP
aceF	Escherichia coli	Ceftazidime	Cephalosporin (3rd)	0.01	1.00	0.88	1.00	0.88	0.01	SNP
aceF	Escherichia coli	Ceftriaxone	Cephalosporin (3rd)	0.01	1.00	0.83	1.00	0.83	0.03	SNP
aceF	Escherichia coli	Cefuroxime	Cephalosporin (2nd)	0.00	1.00	0.82	1.00	0.82	0.03	SNP
aceF	Escherichia coli	Piperacillin-tazobactam	Penicillin	0.01	1.00	0.88	1.00	0.88	0.01	SNP
aceF	Escherichia coli	Trimethoprim-sulfamethoxazole	Folate pathway inhibitors	0.00	1.00	0.68	1.00	0.68	0.10	SNP

Figure 17.4: *A Performance Table.*

- **Gene Name**. The name of the resistance marker evaluated in the light of antimicrobial susceptibility test data. Note, that these can be either complete gene markers or single-nucleotide polymorphisms (see the Type column).
- **Species**. The name of the species used in the antimicrobial susceptibility test.
- **Compound**. The name of the compound used in the antimicrobial susceptibility test.
- **Compound Class**. The compound class for the compound these statistics are calculated for.
- **Sensitivity**. The number of resistant isolates with the marker divided by the total number of resistant isolates in the confusion matrix.
- **Specificity**. The number of susceptible isolates without the marker divided by the total number of susceptible isolates in the confusion matrix.
- **Accuracy**. The (Number of resistant isolates with the marker + Number of susceptible isolates without the marker) divided by the total number of isolates.
- **ppv**. The positive predictive value is calculated as the number resistant isolates with the marker divided by the number of isolates with the marker.
- **npv**. The negative predictive value is calculated as the number of susceptible isolates without the marker divided by the number of isolates without the marker.
- **P-value**. A P-value calculated using Fisher's exact test based on the confusion matrix numbers.
- **Type**. This indicates whether it is a gene coding sequence (CDS) or a single-nucleotide polymorphism (SNP) marker.

The table with performance data is synchronized with the protein table and a selection made in the former will trigger a selection in the latter.

The ARES Database Overview Report The ARES database overview contains some summary statistics on the content of the ARES database (figure 17.5).



1 Ares Database Overview

Number Of Species	Number Of Isolates	Number Of Gene Markers	Number Of SNP Markers	Number Of Compounds	Number Of Associations	Number Of Significant Associations (P-values < 0.05)
111	11059	2634	746	25	49243	26083

2 Species Overview

Species Name	Number Of Isolates	Number Of SNPs	Number Of Genes	Number Of Associations	Number Of Significant Associations (P-values < 0.05)
Acinetobacter baumannii	411	82	170	2921	734
Acinetobacter beijerinckii	2	0	0	20	0
Acinetobacter bereziniae	11	0	0	6	0
Acinetobacter calcoaceticus	0	0	0	313	0
Acinetobacter gyllenbergii	4	0	0	22	0
Acinetobacter haemolyticus	19	0	0	43	0
Acinetobacter indicus	1	0	0	14	0
Acinetobacter johnsonii	9	0	0	6	2
Acinetobacter junii	42	0	0	46	0
Acinetobacter lactucae	2	0	0	192	0
Acinetobacter lwoffii	29	0	0	25	1
Acinetobacter nosocomialis	21	0	0	300	18
Acinetobacter oleivorans	4	0	0	198	0

Figure 17.5: *An example of ARES Database Overview Report.*

- **Number of Isolates.** The number of isolates that have been experimentally tested for resistance against multiple compounds
- **Number of SNPs.** The number of single-nucleotide polymorphisms (corresponding to the entries in the Point Mutation Marker Table)
- **Number of Genes.** The number of gene markers (corresponding to the entries in the Protein Marker Table)
- **Number of Associations.** This is the number of entries in the ARES Performance tables described in the previous section
- **Number of Significant Associations (P-values < 0.05).** The number of performance table entries where the Fisher's exact test P-value was less than 5%. The values are not corrected for multiple testing.

Part VII

Panel Support

Chapter 18

QIAseq 16S/ITS Demultiplexer

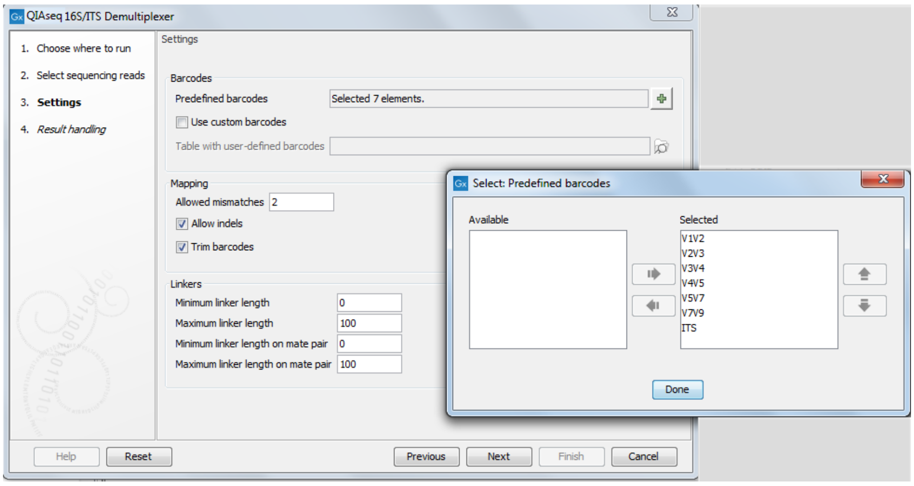
The Panel Support section offers a tool to demultiplex NGS reads of different bacterial variable and fungal ITS regions obtained with the QIAGEN QIAseq 16S/ITS Screening and Region panels. Using this tool, sequences are associated with a particular region when they contain a match to a particular barcode. Sequences that do not contain a match to any of the barcode sequences provided are classified as not grouped.

To run the tool, go to:

> **Microbial Genomics Module** ( ) | **Panel Support** ( ) | **QIAseq 16S/ITS Demultiplexer** ( )

In the first dialog, window select the reads you wish to demultiplex and click **Next**. It is possible to run the tool in batch mode.

In the second dialog, you can choose barcodes, i.e., the stretch of nucleotides used to demultiplex the sequences, from a predefined list (see the list in figure 18.1) or from a custom list.



QIAseq 16S/ITS Demultiplexer

1. Choose where to run
2. Select sequencing reads
3. **Settings**
4. *Result handling*

Settings

Barcodes

- Predefined barcodes: Selected 7 elements.
- ☐ Use custom barcodes
- Table with user-defined barcodes:

Mapping

- Allowed mismatches: 2
- ☑ Allow indels
- ☑ Trim barcodes

Linkers

- Minimum linker length: 0
- Maximum linker length: 100
- Minimum linker length on mate pair: 0
- Maximum linker length on mate pair: 100

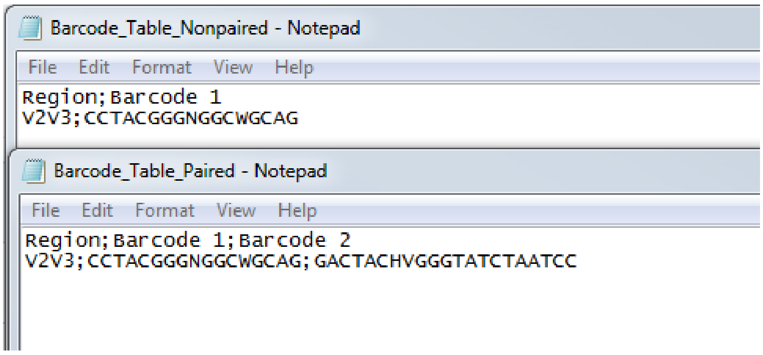
Select: Predefined barcodes

Available:

Selected: V1V2, V2V3, V3V4, V4V5, V5V7, V7V9, ITS

Figure 18.1: *Set the parameters to demultiplex.*

If you choose to use a table of custom barcodes (figure 18.2), you need to specify an Excel or a CSV file previously saved in the Navigation Area. The table will be different when setting barcodes for single or paired reads: for single reads, the first column defines the barcode name, the second contains the barcode sequence. For paired reads, an additional third column contains the reverse complement of the barcode sequence.


Barcode_Table_Nonpaired - Notepad
File Edit Format View Help

```
Region;Barcode 1
V2V3;CCTACGGGNGGCWGCAG
```

Barcode_Table_Paired - Notepad
File Edit Format View Help

```
Region;Barcode 1;Barcode 2
V2V3;CCTACGGGNGGCWGCAG;GACTACHVGGGTATCTAATCC
```

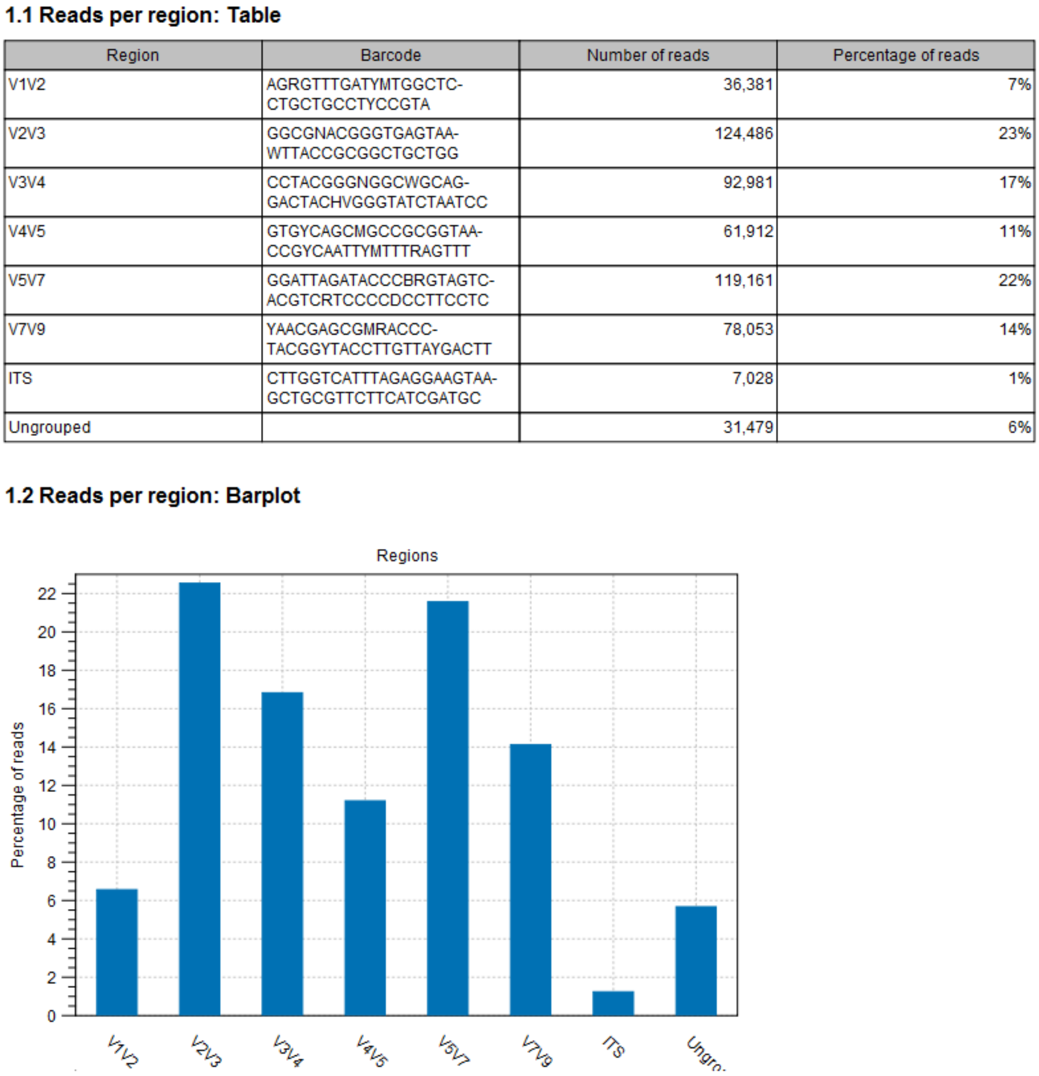
Figure 18.2: *Examples of CSV custom barcodes files for paired and single reads.*

The following parameters for demultiplexing are also available in this dialog:

- Mapping
  - Allow mismatches: decide how many mismatches are allowed between the sequence and the barcode
  - Allow indels
  - Trim barcodes
- Linkers: also known as adapters, linkers are sequences which should just be ignored - it is neither the barcode nor the sequence of interest. For this element, you simply define its length.
  - Minimum linker length
  - Maximum linker length
  - Minimum linker length on mate pair
  - Maximum linker length on mate pair

In the Result handling window, you can choose to Create a report (see figure 18.3), and Save a sequence list of all ungrouped sequences.

The main output are sequence lists for each different regions/barcodes. These sequence lists can be used as input for the Data QC and OTU Clustering workflow, that will generate an output table displaying the OTUs abundances for each region. Note that the Trim Reads step of the workflow will automatically detect and trim the remaining read-through barcodes found on paired-end reads and not discarded by the demultiplexer. However, if you are working with single reads, mate-paired reads or data of low quality, it is recommended to specify a trim adapter list containing all barcodes in the Trim Reads step of the workflow.



1.1 Reads per region: Table

Region	Barcode	Number of reads	Percentage of reads
V1V2	AGRGTTTGATYMTGGCTC-CTGCTGCCTYCCGTA	36,381	7%
V2V3	GGCGNACGGGTGAGTAA-WTTACCGCGGCTGCTGG	124,486	23%
V3V4	CCTACGGGNGGCWGCAG-GACTACHVGGGTATCTAATCC	92,981	17%
V4V5	GTGYCAGCMGCCGCGGTAA-CCGYCAATTYMTTTRAGTTT	61,912	11%
V5V7	GGATTAGATACCCBRGTAGTC-ACGTCRTCCCCDCCTTCCTC	119,161	22%
V7V9	YAACGAGCGMRACCC-TACGGYTACCTTGTTAYGACTT	78,053	14%
ITS	CTTGGTCATTTAGAGGAAGTAA-GCTGCGTTCTTCATCGATGC	7,028	1%
Ungrouped		31,479	6%

1.2 Reads per region: Barplot

Figure 18.3: *An example of demultiplexing report.*

Part VIII

Utility Tools

Chapter 19

Utility Tools

19.1 Mask Low-Complexity Regions

The **Mask Low-Complexity Regions** tool can be used to identify and mask repetitive regions in sequences. In some cases this can remove erroneous matches: for instance, when doing taxonomic profiling, a read with a highly repetitive sequence is likely to match a reference genome purely by chance.

The tool takes any sequence or sequence list as input (including reads and genomes). It will accept both nucleotide and protein sequence input.

To run the tool, go to

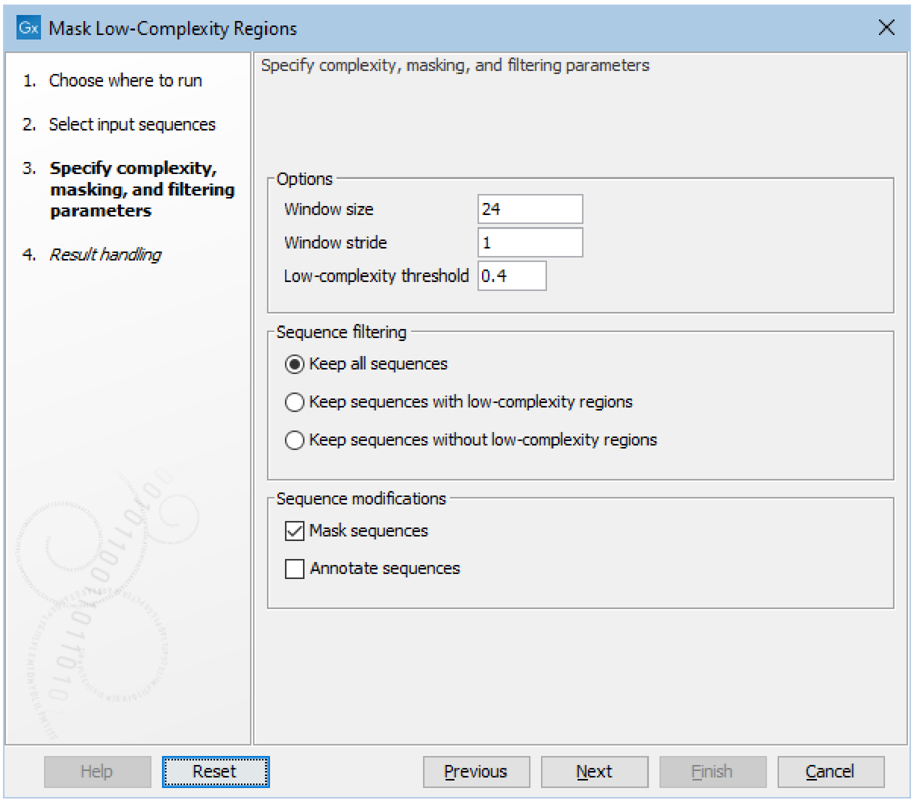
> **Toolbox** | **Utility Tools** ( ) | **Mask Low-Complexity Regions ( )**

The following general options are available (figure 19.1):

- **Window size**: The complexity is evaluated by moving a window along the sequences. This option sets the sliding window size.
- **Window stride**: The number of nucleotides by which the window is moved along the sequence. Increasing this value makes the tool faster, but slightly less accurate when detecting the edges of low-complexity regions.
- **Low-complexity threshold**: This measure is normalized such that a value of 0 corresponds to a trivial sequence (e.g. 'AAAAAAAA'), and 1 corresponds to a random sequence. Higher values mask more of the sequence. Notice, that the report contains sequence examples for different complexity thresholds.

The Sequence filtering options make it possible to specify whether some or all input sequences should be output:

- **Keep all sequences**. No filtering is performed.
- **Keep sequences with low-complexity regions**. Notice, that sequences which have already been masked in a previous run of the tool will not be kept.
- **Keep sequences without low-complexity regions**. Note, that this also keeps sequences where low complexity regions have already been masked in a previous run of the tool.



Mask Low-Complexity Regions

1. Choose where to run
2. Select input sequences
3. **Specify complexity, masking, and filtering parameters**
4. *Result handling*

Specify complexity, masking, and filtering parameters

Options

Window size: 24

Window stride: 1

Low-complexity threshold: 0.4

Sequence filtering

● Keep all sequences

○ Keep sequences with low-complexity regions

○ Keep sequences without low-complexity regions

Sequence modifications

☑ Mask sequences

□ Annotate sequences

Figure 19.1: *The Mask Low-Complexity Regions options.*

Finally, the Sequence modifications options determine how the output sequences are marked:

- **Mask sequences**: Low-complexity regions are masked with N's (or X's for proteins). Notice, that if a tested window already contains ambiguous symbols (e.g. from a previous run of the tool), it will not be masked.

- **Annotate sequences**: Low-complexity regions are marked with a sequence annotation. Notice, that if a tested window already contains ambiguous symbols (e.g. from a previous run of the tool), it will not be marked.

The tool optionally outputs a report with statistics on the detected regions. The report is described in details below:

19.1.1 Mask Low-Complexity Regions Report

An example of a Mask Low-Complexity Regions report can be seen in figure 19.2.

The summary statistics describes the following measures:

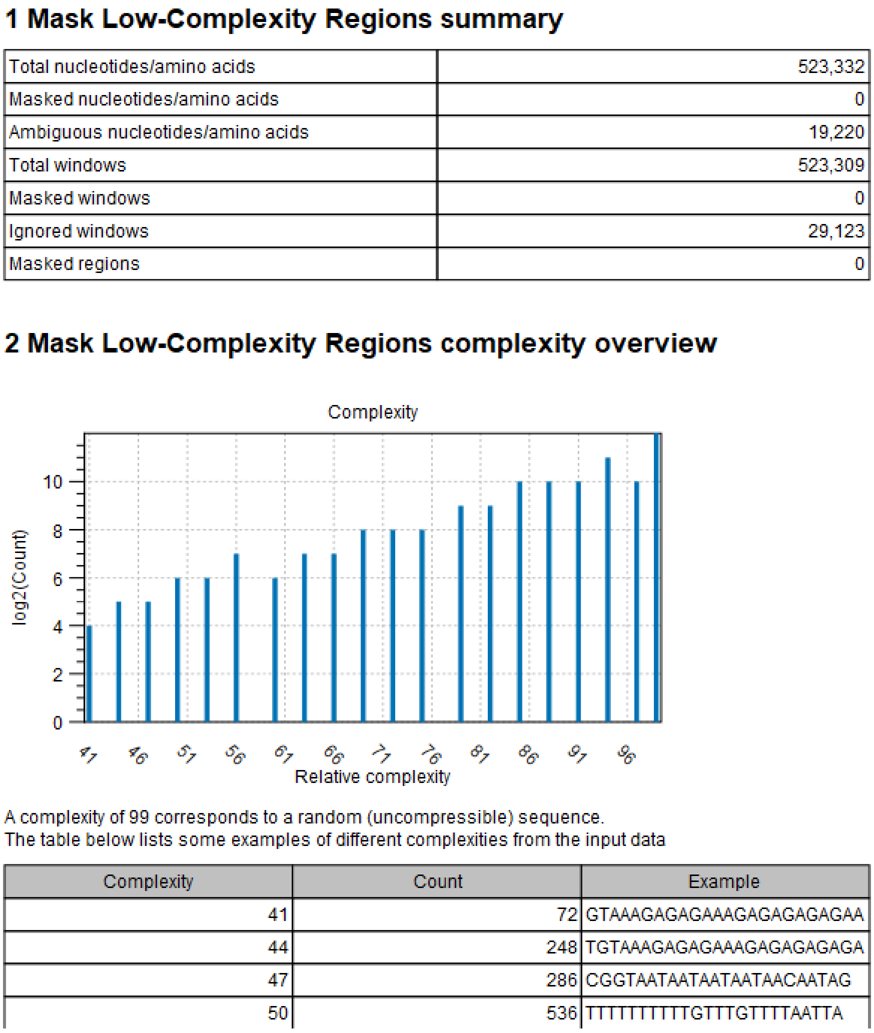
- **Total nucleotides/amino acids**: The total number of nucleotides (or amino acids for protein sequence input) that were processed.

- **Masked nucleotides/amino acids**: The total number of nucleotides (or amino acids for protein sequence input) that were masked or annotated.

- **Ambiguous nucleotides/amino acids**: The total number of ambiguous nucleotides (or amino acids for protein sequence input) that were masked or annotated.
- **Total windows**: The total number of windows that were processed (notice, that these may be overlapping)
- **Masked windows**: The total number of windows that were masked (notice, that these may be overlapping)
- **Ignored windows**: The total number of windows that were ignored (because they already contained ambiguous nucleotides, e.g. from an earlier run)
- **Masked regions**: The total number of regions that were masked (regions are formed by joining all overlapping or adjacent windows into contiguous sections)

The Mask Low-Complexity Regions complexity overview shows a bar chart with the number of windows for the different complexity levels.

A table provides examples from the input data for different complexity levels: this is shown in order to make it easier to understand what the **Low-complexity threshold** corresponds to.



1 Mask Low-Complexity Regions summary

Total nucleotides/amino acids	523,332
Masked nucleotides/amino acids	0
Ambiguous nucleotides/amino acids	19,220
Total windows	523,309
Masked windows	0
Ignored windows	29,123
Masked regions	0

2 Mask Low-Complexity Regions complexity overview

A complexity of 99 corresponds to a random (uncompressible) sequence.
The table below lists some examples of different complexities from the input data

Complexity	Count	Example
41	72	GTAAAGAGAGAAAGAGAGAGAGAA
44	248	TGTAAAGAGAGAAAGAGAGAGAGA
47	286	CGGTAATAATAATAATAACAATAG
50	536	TTTTTTTTTTGTTTGTTTTAATTA

Figure 19.2: *The Mask Low-Complexity Regions report.*

19.2 Result Metadata

Metadata refers to information about data. In the context of the CLC Microbial Genomics Module, this usually means information about samples. For example a set of reads could come from a particular specimen at a particular time point with particular characteristics. The specimen, time and characteristics would be metadata for that set of reads.

What is metadata used for?

Core uses of metadata in CLC software include:

- Defining batch units when launching workflows in batch mode, described in https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Running_workflows_in_batch_mode.html.
- Distributing data to the relevant input channels in a workflow when using Collect and Distribute elements, described in https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Control_flow_elements.html.
- Finding and selecting data elements based on sample information (in a CLC Metadata Table). Workflow Result Metadata Tables are of particular use when reviewing results generated by workflows run in batch mode and are described in https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Workflow_Result_Metadata_tables.html.
- Running tools where characteristics of the data elements are relevant. An example is Differential Abundance Analysis, described in section 6.6.

Metadata tables

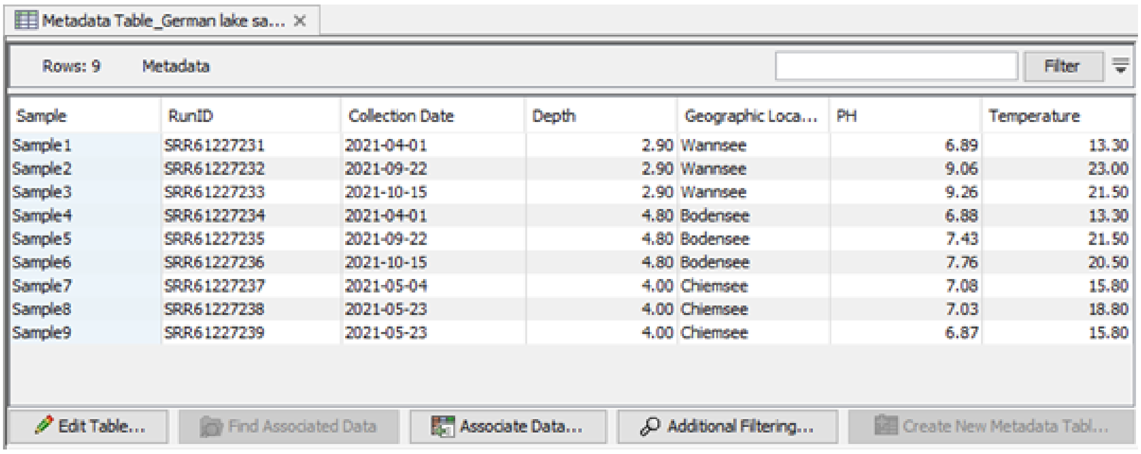
An example of a CLC Metadata Table in the CLC Microbial Genomics Module is shown in figure 19.3. Each column represents a property of a sample (e.g., identifier, sample depth, geographic location, temperature) and each row contains information relevant to a sample. A single column can be designated the key column. That column must contain unique entries.

Each row can have associations with one or more data elements, such as sequence lists, taxonomic profiling abundance tables, variant tracks, etc.

Creating metadata tables

CLC Metadata Tables can be created in several ways, including:

- Import metadata from an Excel, CSV or TSV format file using the **Import Metadata** ( ) tool. You can associate already imported data with your metadata during import, or do this later. The process of importing metadata and associating data is described in the CLC Genomics Workbench user manual, https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Importing_metadata.html.
- Use a workflow to import a sample and its metadata at the same time. A template workflow for importing sequence data with associated metadata can be found in the Preparing Raw



Sample	RunID	Collection Date	Depth	Geographic Loca...	PH	Temperature
Sample1	SRR61227231	2021-04-01	2.90	Wannsee	6.89	13.30
Sample2	SRR61227232	2021-09-22	2.90	Wannsee	9.06	23.00
Sample3	SRR61227233	2021-10-15	2.90	Wannsee	9.26	21.50
Sample4	SRR61227234	2021-04-01	4.80	Bodensee	6.88	13.30
Sample5	SRR61227235	2021-09-22	4.80	Bodensee	7.43	21.50
Sample6	SRR61227236	2021-10-15	4.80	Bodensee	7.76	20.50
Sample7	SRR61227237	2021-05-04	4.00	Chiemsee	7.08	15.80
Sample8	SRR61227238	2021-05-23	4.00	Chiemsee	7.03	18.80
Sample9	SRR61227239	2021-05-23	4.00	Chiemsee	6.87	15.80

Figure 19.3: *A CLC Metadata Table, with the key column highlighted in blue.*

Data folder in the Template Workflows section of the Toolbox. The template is described in the CLC Genomics Workbench user manual, https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Import_with_Metadata.html.

For more ways to create CLC Metadata Tables and information on how to work with CLC Metadata Tables in general, see the Metadata section of the CLC Genomics Workbench user manual, https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Metadata.html.

In addition to the CLC Metadata Table, the CLC Microbial Genomics Module makes use of a special type of metadata table; the Result Metadata Table. As opposed to the CLC Metadata Table, the Result Metadata Table can be updated with selected types of analysis results e.g., antibiotic resistance. This is described in section 19.2.1.

19.2.1 Create a Result Metadata Table

A Result Metadata Table is generated from a CLC Metadata Table with associated sample data.

To run the tool, go to:

Toolbox | **Utility Tools** ( ) | **Result Metadata** ( ) | **Create Result Metadata Table** ( )

Select the CLC Metadata Table (figure 19.4) and click on **Next** to specify result handling.

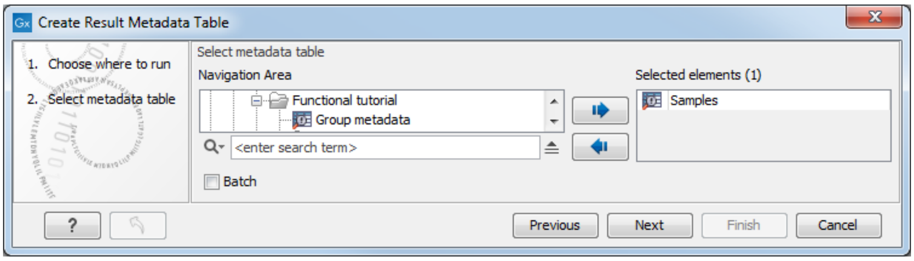


Figure 19.4: *Creation of a Result Metadata Table from a Metadata Table.*

The tool outputs a Result Metadata Table.

When first opened, the Result Metadata Table is empty (figure 19.5).

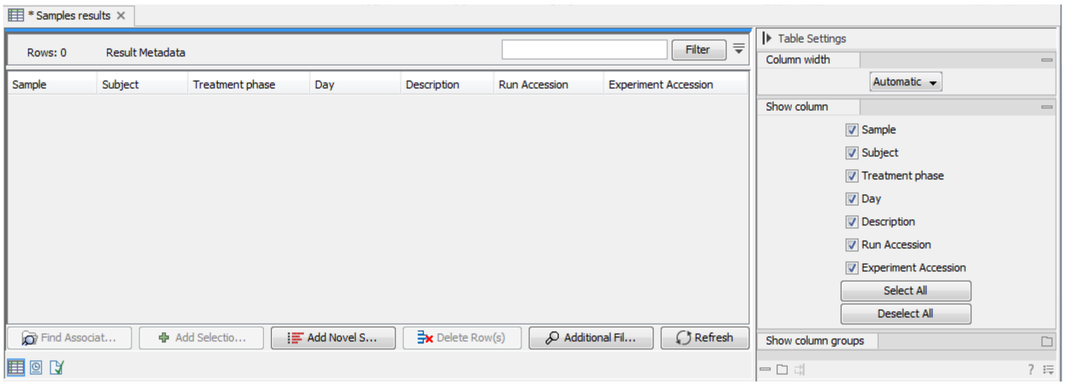
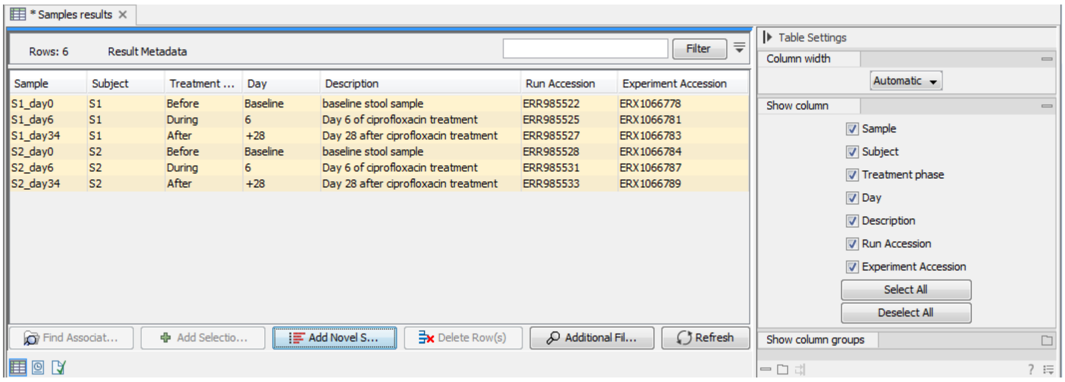


Figure 19.5: *The newly created Result Metadata Table is empty.*

To populate the table with information from the underlying CLC Metadata Table, click on **Add Novel Samples** (). Samples and associated metadata will be listed marked in yellow (figure 19.6). Save the Result Metadata Table to store the sample and metadata information.



Sample	Subject	Treatment ...	Day	Description	Run Accession	Experiment Accession
S1_day0	S1	Before	Baseline	baseline stool sample	ERR985522	ERX1066778
S1_day6	S1	During	6	Day 6 of ciprofloxacin treatment	ERR985525	ERX1066781
S1_day34	S1	After	+28	Day 28 after ciprofloxacin treatment	ERR985527	ERX1066783
S2_day0	S2	Before	Baseline	baseline stool sample	ERR985528	ERX1066784
S2_day6	S2	During	6	Day 6 of ciprofloxacin treatment	ERR985531	ERX1066787
S2_day34	S2	After	+28	Day 28 after ciprofloxacin treatment	ERR985533	ERX1066789

Figure 19.6: *Click on "Add Novel Samples" to add metadata from the underlying CLC Metadata Table to the otherwise empty Result Metadata Table.*

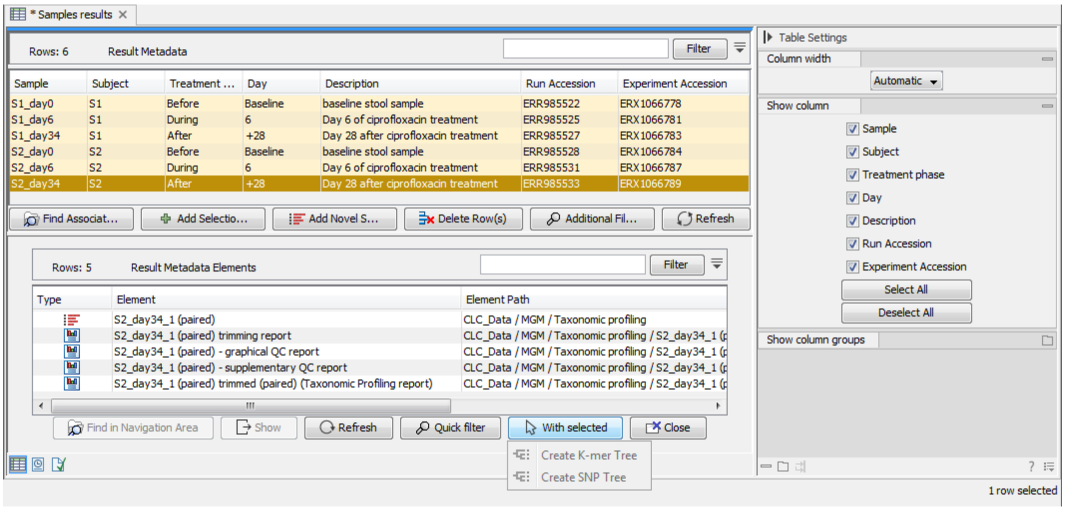
If for some reason Result Metadata rows are not needed, they can be deleted from the table by selecting them and clicking on the **Delete Row(s)** () button.

To find files associated to specific Metadata rows, select the sample row(s) of interest and click on **Find Associated Data** (). This action will list all associated files in a new Metadata Element window located below the Metadata window as shown in figure 19.7.

In most cases, analysis results will be added automatically to the Result Metadata Table when using a properly designed workflow. It is also possible to add manually generated analysis results to the table using the **Extend Result Metadata Table**.

19.2.2 Running an analysis directly from a Result Metadata Table

Analysis results from tools listed in the table of section 19.2.1 are automatically added to the Result Metadata Table as long as it was performed on samples associated with metadata. Content of the Result Metadata Table may be managed in similar ways as other tables in CLC Genomics Workbench (`https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Filtering_tables.html`), but it can also be used to start new analyses using the **With selected** ( ) button which provides the option of various downstream analysis of



Rows: 6 Result Metadata

Sample	Subject	Treatment ...	Day	Description	Run Accession	Experiment Accession
S1_day0	S1	Before	Baseline	baseline stool sample	ERR985522	ERX1066778
S1_day6	S1	During	6	Day 6 of ciprofloxacin treatment	ERR985525	ERX1066781
S1_day34	S1	After	+28	Day 28 after ciprofloxacin treatment	ERR985527	ERX1066783
S2_day0	S2	Before	Baseline	baseline stool sample	ERR985528	ERX1066784
S2_day6	S2	During	6	Day 6 of ciprofloxacin treatment	ERR985531	ERX1066787
S2_day34	S2	After	+28	Day 28 after ciprofloxacin treatment	ERR985533	ERX1066789

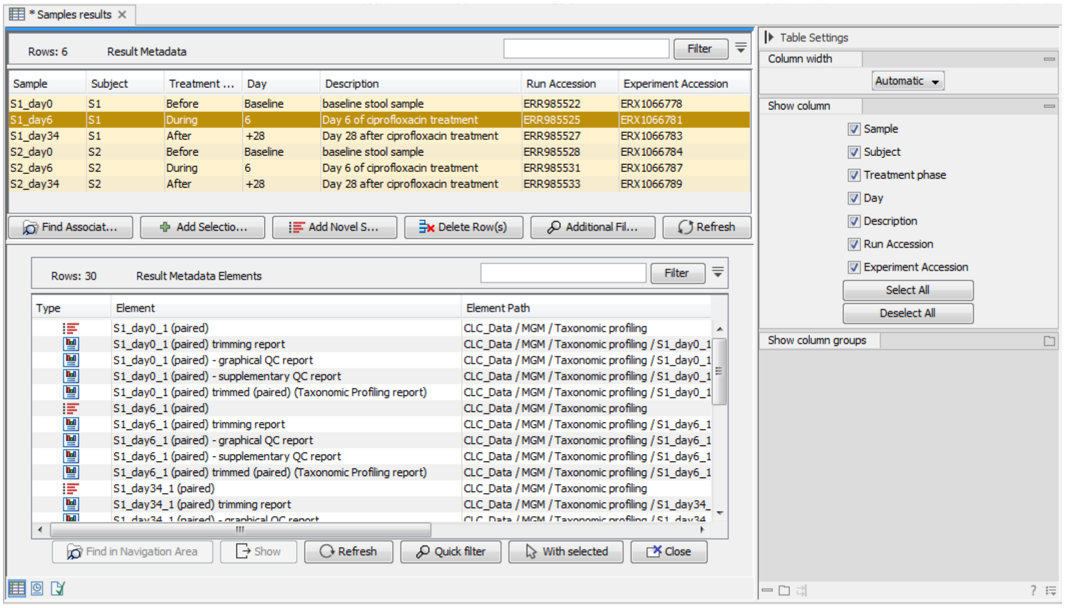
Rows: 5 Result Metadata Elements

Type	Element	Element Path
☰	S2_day34_1 (paired)	CLC_Data / MGM / Taxonomic profiling
▦	S2_day34_1 (paired) trimming report	CLC_Data / MGM / Taxonomic profiling / S2_day34_1 (p
▦	S2_day34_1 (paired) - graphical QC report	CLC_Data / MGM / Taxonomic profiling / S2_day34_1 (p
▦	S2_day34_1 (paired) - supplementary QC report	CLC_Data / MGM / Taxonomic profiling / S2_day34_1 (p
▦	S2_day34_1 (paired) trimmed (paired) (Taxonomic Profiling report)	CLC_Data / MGM / Taxonomic profiling / S2_day34_1 (p

Figure 19.7: *The Metadata Element window at the bottom part of this figure lists all data associated to the selected Result Metadata row shown in the top window. In this example, only the imported read file is associated to the single metadata row. Note! Workflow analysis can be initiated directly on "With selected" Elements.*

the selected dataset.

To perform an analysis on one or more samples, begins by selecting the relevant rows followed by finding the associated elements by clicking on the **Find Associated Data** ( ) button. All associated elements are then listed in window below called **Metadata Elements**. You can see an example in figure 19.8, where a Metadata Result Table includes 6 rows (Metadata, top view), while 30 elements are found to be associated to these 6 rows (Metadata Elements, bottom view).



Rows: 6 Result Metadata

Sample	Subject	Treatment ...	Day	Description	Run Accession	Experiment Accession
S1_day0	S1	Before	Baseline	baseline stool sample	ERR985522	ERX1066778
S1_day6	S1	During	6	Day 6 of ciprofloxacin treatment	ERR985525	ERX1066781
S1_day34	S1	After	+28	Day 28 after ciprofloxacin treatment	ERR985527	ERX1066783
S2_day0	S2	Before	Baseline	baseline stool sample	ERR985528	ERX1066784
S2_day6	S2	During	6	Day 6 of ciprofloxacin treatment	ERR985531	ERX1066787
S2_day34	S2	After	+28	Day 28 after ciprofloxacin treatment	ERR985533	ERX1066789

Rows: 30 Result Metadata Elements

Type	Element	Element Path
☰	S1_day0_1 (paired)	CLC_Data / MGM / Taxonomic profiling
▦	S1_day0_1 (paired) trimming report	CLC_Data / MGM / Taxonomic profiling / S1_day0_1
▦	S1_day0_1 (paired) - graphical QC report	CLC_Data / MGM / Taxonomic profiling / S1_day0_1
▦	S1_day0_1 (paired) - supplementary QC report	CLC_Data / MGM / Taxonomic profiling / S1_day0_1
▦	S1_day0_1 (paired) trimmed (paired) (Taxonomic Profiling report)	CLC_Data / MGM / Taxonomic profiling / S1_day0_1
☰	S1_day6_1 (paired)	CLC_Data / MGM / Taxonomic profiling
▦	S1_day6_1 (paired) trimming report	CLC_Data / MGM / Taxonomic profiling / S1_day6_1
▦	S1_day6_1 (paired) - graphical QC report	CLC_Data / MGM / Taxonomic profiling / S1_day6_1
▦	S1_day6_1 (paired) - supplementary QC report	CLC_Data / MGM / Taxonomic profiling / S1_day6_1
▦	S1_day6_1 (paired) trimmed (paired) (Taxonomic Profiling report)	CLC_Data / MGM / Taxonomic profiling / S1_day6_1
☰	S1_day34_1 (paired)	CLC_Data / MGM / Taxonomic profiling
▦	S1_day34_1 (paired) trimming report	CLC_Data / MGM / Taxonomic profiling / S1_day34_
▦	S1_day34_1 (paired) - graphical QC report	CLC_Data / MGM / Taxonomic profiling / S1_day34

Figure 19.8: *In total, 30 files are associated to the selected 6 sample rows within the Result Metadata Table.*

As the number of samples, metadata and data elements increases over time, and the Result Metadata Table likely will include a mix of analyzed and novel samples, it is helpful to perform filtering steps to identify the elements you are looking for (see section 19.2.2). Once filtering is done, it is easy to select the remaining rows of data elements and click the **With selected** (⇖) button to start tools such as **Create K-mer Tree** and **Create SNP Tree**, or initiate a workflow analyses using an opened and customized version of a workflow.

Filtering in Result Metadata Table

Filtering is generally performed as a two step process: by picking or filtering firstly on the rows of the Result Metadata Table and secondly among the associated Metadata Elements.

Filtering can be done several ways, usually using a combination of the following options:

- Use the traditional table filtering function in top right corner. Filter for text elements, or unroll the banner by clicking on the icon (▼) and use more specific filters options.
- Tables can be sorted according to one or more columns, making it easier to find (and select) the desired elements. One example is to click on the **Role** column to find data elements with the same role.
- In the case of Metadata elements, use the **Quick filter** (🔍) button and select the desired filtering option. It is possible to choose among:
  - **Imported** filters down to elements with the "Role" being **Sample data**. This can for instance be used for analyzing using an open and validated workflow based on one of the template workflows from the Toolbox by clicking the **With selected** (⇖) button.
  - **Filter for SNP Tree** filters down to elements with the "Role" being either **Read mapping**, **Realigned mapping** or **Variants**. Selection of the elements remaining after this filtering has been applied makes it easy to click the **With selected** (⇖) button and initiate the **Create SNP Tree** tool using the selected data as input.
  - **Filter for K-mer Tree** filters down to elements with the "Role" being **Trimmed reads**. Selection of the elements remaining after this filtering has been applied makes it easy to click the **With selected** (⇖) button and initiate the **Create K-mer Tree** tool using the selected data as input.
  - **Filter Re-mapped 'name of common reference' for SNP Tree**, option available for elements generated with the **Map to Specified Reference** or manually added with the **Use Genome as Result** and based on a shared reference.

  Applied quick filter selection can be removed by clicking the **Quick filter** (🔍) followed by **Clear Quick Filter**.

Filtering in a SNP-Tree creation scenario

To construct a SNP tree, all sample data must have been analyzed (i.e., reads mapped and variants called) using the same reference sequence. If we want to use all the samples that were generated by the Map to Specified Reference workflow on several occasions using a common reference sequence, we use the quick filtering options.

- **Filter** all samples where read mapping and variant calling was performed using a common reference by clicking on the icon (▼) and using the following filter parameters: in the first drop-down menu, choose the column whose header is the reference sequence of interest; in the second drop-down menu, choose the term "contains"; and in the third window, write "true".

- Select all remaining samples.
- **Click** on the **Find Associated Data** () button. This opens the Metadata elements table underneath the initial Metadata table with a certain amount of elements associated with the samples selected in the Metadata Result Table.
- **Click** on the **Quick Filters** () button in the Metadata Elements Table (bottom view) and select the **Filter Re-mapped 'common reference' for SNP Tree** option.
- **Select** all the remaining elements.
- **Click** on the **With selected** ( ) button and select the **Create SNP Tree** option. The **Create SNP Tree** wizard is displayed (see section 9.1). The read mappings are preselected as input. The variant tracks and the Result Metadata Table are automatically preselected as parameters.

19.2.3 Extend Result Metadata Table

The **Extend Result Metadata Table** tool adds one or more Result objects to the Result Metadata Table. The tool outputs a copy of the source Result Metadata Table. The original source table is not modified.

To manually add results to an existing Result Metadata Table, go to:

Toolbox | **Utility Tools** () | **Result Metadata** () | **Extend Result Metadata Table** ()

1. In the first wizard window, select the relevant Result Metadata Table (see figure 19.9) and click **Next**.

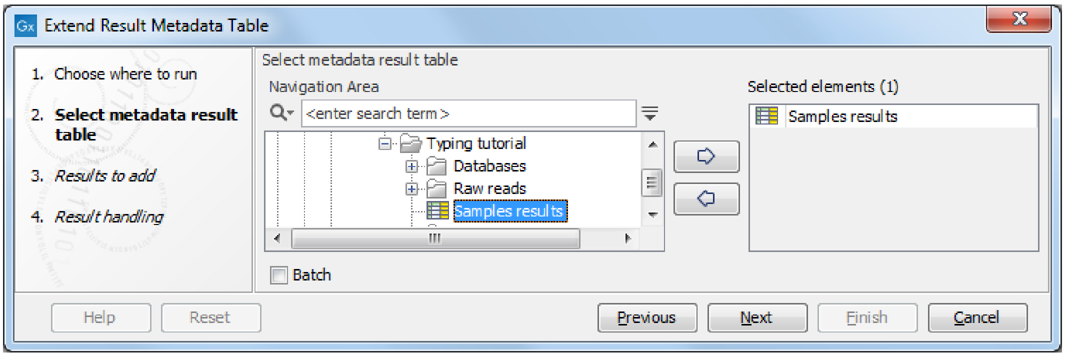
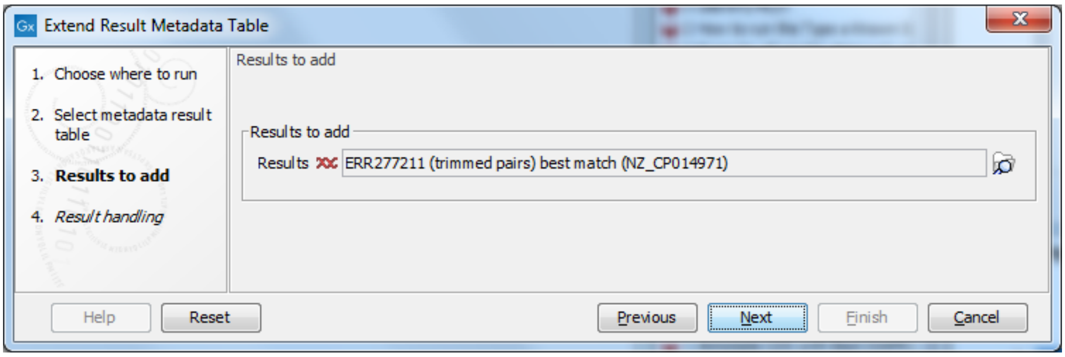


Figure 19.9: *First select the Result Metadata Table you want to add results to.*

2. Now select the relevant Result object(s) to be added to the Result Metadata Table (see figure 19.10) and click **Next**.
3. Finally, select to **Save** and click **Finish**.

The output of this tool is a copy of the Result Metadata Table containing cells updated with the results (figure 19.11).



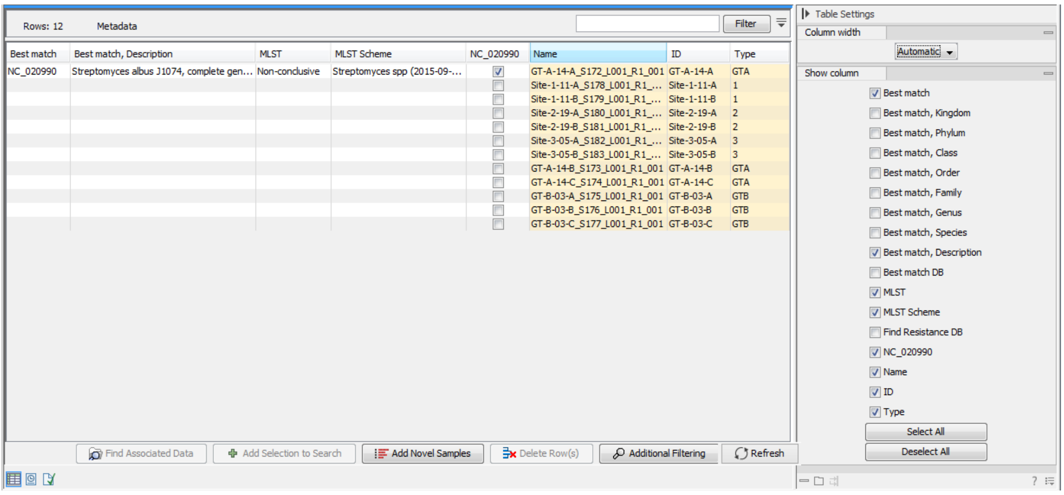
1. Choose where to run
2. Select metadata result table
3. **Results to add**
4. *Result handling*

Results to add

Results to add

Results: ERR277211 (trimmed pairs) best match (NZ_CP014971)

Figure 19.10: *In the second step of this example, the identified best matching sequence for a particular sample is added.*



Rows: 12 Metadata

Best match	Best match, Description	MLST	MLST Scheme	NC_020990	Name	ID	Type
NC_020990	Streptomyces albus J1074, complete gen...	Non-conclusive	Streptomyces spp (2015-09-...	☑	GT-A-14-A_S172_L001_R1_001	GT-A-14-A	GTA
				□	Site-1-11-A_S178_L001_R1_...	Site-1-11-A	1
				□	Site-1-11-B_S179_L001_R1_...	Site-1-11-B	1
				□	Site-2-19-A_S180_L001_R1_...	Site-2-19-A	2
				□	Site-2-19-B_S181_L001_R1_...	Site-2-19-B	2
				□	Site-3-05-A_S182_L001_R1_...	Site-3-05-A	3
				□	Site-3-05-B_S183_L001_R1_...	Site-3-05-B	3
				□	GT-A-14-B_S173_L001_R1_001	GT-A-14-B	GTA
				□	GT-A-14-C_S174_L001_R1_001	GT-A-14-C	GTA
				□	GT-B-03-A_S175_L001_R1_001	GT-B-03-A	GTB
				□	GT-B-03-B_S176_L001_R1_001	GT-B-03-B	GTB
				□	GT-B-03-C_S177_L001_R1_001	GT-B-03-C	GTB

Table Settings

Column width: Automatic

Show column

☑ Best match
□ Best match, Kingdom
□ Best match, Phylum
□ Best match, Class
□ Best match, Order
□ Best match, Family
□ Best match, Genus
□ Best match, Species
☑ Best match, Description
□ Best match DB
☑ MLST
☑ MLST Scheme
□ Find Resistance DB
☑ NC_020990
☑ Name
☑ ID
☑ Type

Figure 19.11: *Example with column added to the Result Metadata Table, including the data for the particular sample that was specified in step 1.*

19.2.4 Use Genome as Result

The **Use Genome as Result** tool is part of the **Map to Specified Reference** workflow scenario and is not necessarily intended to be used as a single tool by users. Its function, at the last step of the **Map to Specified Reference** workflow, is double: it adds the name of the reference genome used for the re-mapping to the 'role' of the input files (for example the role "mapping report" will become "NZ_CP014971 mapping report", where NZ_CP014971 is the name of the reference used to re-map). It also creates an extra column in the Result Metadata Table whose header is the name of the common reference that was used for the re-mapping (here NZ_CP014971). This extra column makes it possible to distinguish between read mappings that were generated at different time points as well as in different runs of the workflow, despite using the **same genome reference**.

The tool can take multiple elements as input, and each will have its metadata role changed to include the name of reference sequence in addition to the original role value. Relevant elements can be selected individually, or you can select folders by right-clicking on the folder value and selecting **Add folder contents** (it will select all elements in that folder), or select folder recursively by right-clicking on the folder value and selecting **Add folder contents (recursively)**. In this last case all elements of the folder, including elements contained in subfolders, will be selected (see figure 19.12).

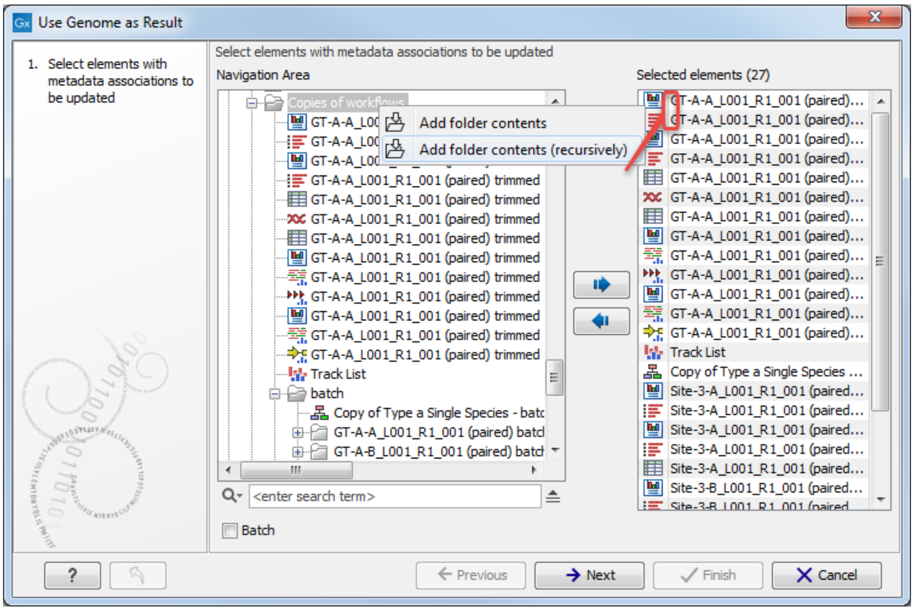
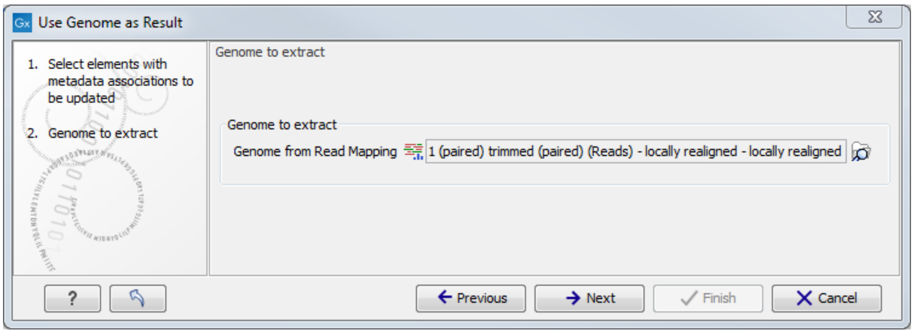

Use Genome as Result
1. Select elements with metadata associations to be updated
Select elements with metadata associations to be updated
Navigation Area
Selected elements (27)
Copies of workflows
Add folder contents
Add folder contents (recursively)
GT-A-A_L001_R1_001 (paired) trimmed
Track List
batch
Copy of Type a Single Species - batc
GT-A-A_L001_R1_001 (paired) batcl
GT-A-B_L001_R1_001 (paired) batcl
<enter search term>
□ Batch
GT-A-A_L001_R1_001 (paired)...
Track List
Copy of Type a Single Species ...
Site-3-A_L001_R1_001 (paired...
Site-3-B_L001_R1_001 (paired...
?
Previous
Next
Finish
Cancel

Figure 19.12: *Select recursively elements with metadata associations to be updated. Here the selected elements are the 16 of the folder 'Copies of workflows' as well as the 9 from the subfolder called 'batch'.*

In the second dialog, select the relevant read mapping, i.e., the read mapping that was created using the common reference you want to annotate the roles with (figure 19.13).



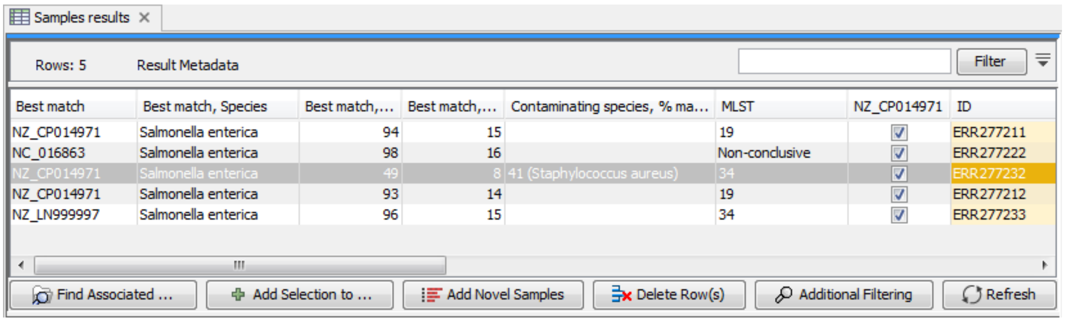
Genome to extract

Genome from Read Mapping: 1 (paired) trimmed (paired) (Reads) - locally realigned - locally realigned

Figure 19.13: *Specification of the read mapping to be associated with genome metadata.*

In addition to changing the role name, the tool creates a new column named after the selected reference sequence in the Result Metadata Table. This column indicates whether data has been analysed using this reference or not (For an example see column "NZ_CP014971" in figure 19.14). To filter for all possible elements that were generated using this sequence as reference data, open the filter banner by clicking on the icon (▼) next to the **Filter** button. In the first drop down menu, choose the column whose header is the reference sequence of interest. In the second drop down menu, choose the term "contains", and in the third window, write "true". This will filter for all the elements with a tick in the reference sequence column, as can be seen on the figure 19.14.

To add the genome output from this tool to a Result Metadata Table, see section 19.2.3.



Rows: 5 Result Metadata

Best match	Best match, Species	Best match,...	Best match,...	Contaminating species, % ma...	MLST	NZ_CP014971	ID
NZ_CP014971	Salmonella enterica	94	15		19	☑	ERR277211
NC_016863	Salmonella enterica	98	16		Non-conclusive	☑	ERR277222
NZ_CP014971	Salmonella enterica	49	8	41 (Staphylococcus aureus)	34	☑	ERR277232
NZ_CP014971	Salmonella enterica	93	14		19	☑	ERR277212
NZ_LN999997	Salmonella enterica	96	15		34	☑	ERR277233

Figure 19.14: *Filter for elements who share a tick in the column newly generated by the Use Genome as Result tool*

Part IX

Legacy tools

Chapter 20

Legacy tools

The documentation in this section is for tools that have been deprecated and that will be retired in a future version. In most cases, deprecated tools can be found in the **Legacy Tools** () folder of the Workbench Toolbox, with "(legacy)" appended to their names to highlight their status.

We recommend redesigning workflows containing any of these tools to remove them. Where a new tool has been introduced to take the deprecated tool's place, please try including the new tool.

If you have concerns about the retirement of particular tools in this section, please contact QIAGEN Bioinformatics Support team at ts-bioinformatics@qiagen.com.

20.1 Extract Regions from Tracks

This tool will be retired in a future version of the software. We recommend using Extract Reads instead, see (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Extract_Reads.html).

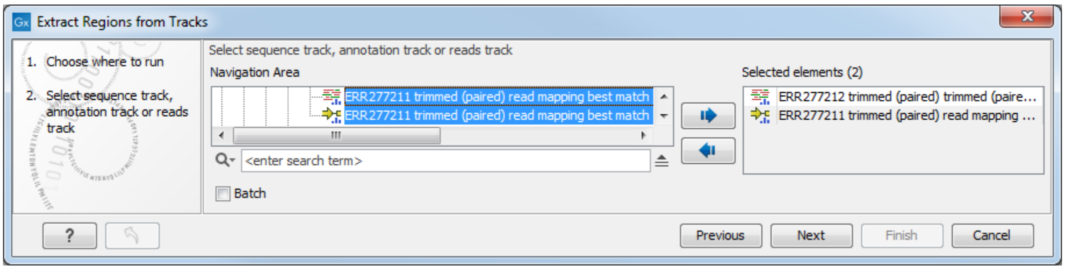
The **Extract Regions from Tracks** tool facilitates specific extraction of mapped reads covering the particular regions specified by the annotation track, e.g., in this case the regions specified by an MLST scheme. The generated Track list enables visualization of the mapped reads per MLST loci. As the tool focuses on visualization of the mapping against the locus, the coverage up- and downstream of the loci does not reflect the actual coverage.

The tool is initiated by:

Toolbox | **Legacy Tools** () | **Extract Regions from Track (legacy)** ()

The input file to the tool is a Reads Track, Genome Track or and/or Annotation Track (figure 20.1) while the specification of the regions to be extracted is specified by the NGS MLST Annotation Track (figure 20.2) generated when initially running the Identify MLST tool.

The extracted mapped reads are visualized in the track list shown in figure 20.3. Through the navigation tool at top right, it is possible to switch among the various MLST regions defined for the analyzed species.



Extract Regions from Tracks

1. Choose where to run
2. Select sequence track, annotation track or reads track

Select sequence track, annotation track or reads track

Navigation Area

- ERR277211 trimmed (paired) read mapping best match
- ERR277211 trimmed (paired) read mapping best match

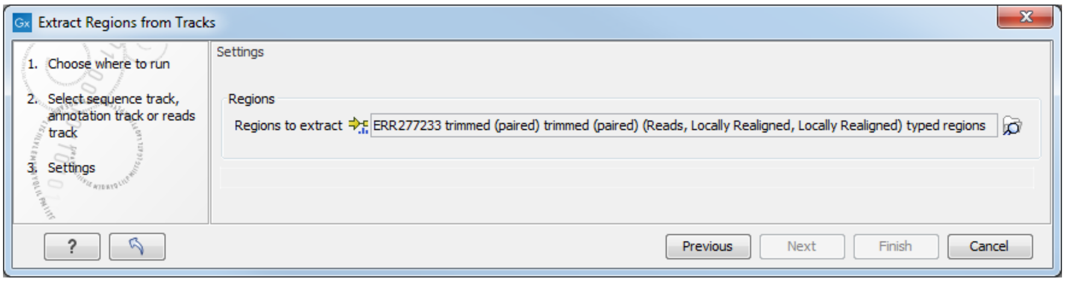
<enter search term>

□ Batch

Selected elements (2)

- ERR277212 trimmed (paired) trimmed (paire...
- ERR277211 trimmed (paired) read mapping ...

Figure 20.1: *The first input file to the Extract Regions from Tracks tool is a Reads Track, Genome Track or and/or Annotation Track.*



Extract Regions from Tracks

1. Choose where to run
2. Select sequence track, annotation track or reads track
3. Settings

Settings

Regions

Regions to extract: ERR277233 trimmed (paired) trimmed (paired) (Reads, Locally Realigned, Locally Realigned) typed regions

Figure 20.2: *The second input file to the Extract Regions from Tracks tool is a NGS-MLST Annotation Track generated by the Identify MLST tool.*

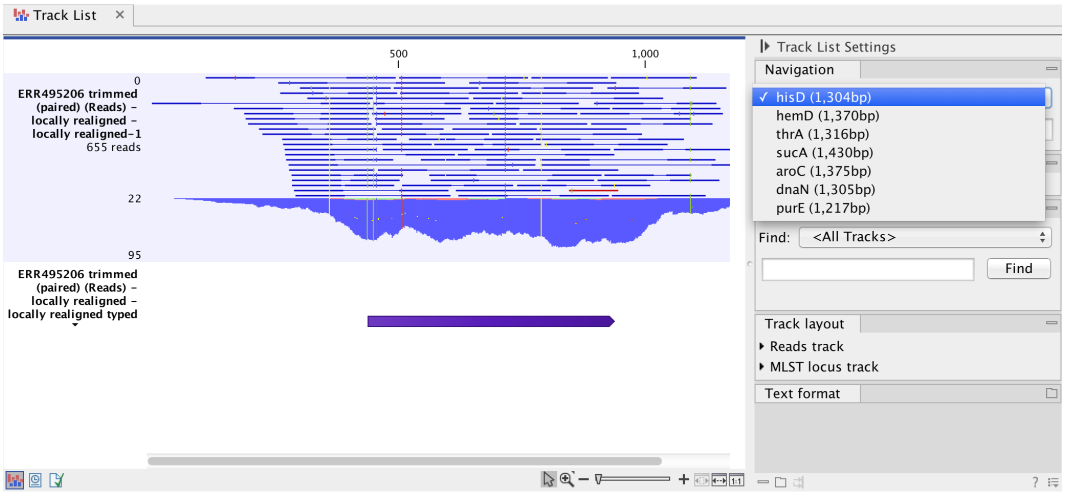


Figure 20.3: *A track list showing the mapped reads covering a specific region defined by the species specific MLST scheme. At top right, the drop down list shows that the hisD region is currently visualized region.*

20.2 Analyze Viral Hybrid Capture Panel Data

This template workflow has been replaced by **Analyze QIAseq xHYB Viral Panel Data (Human host)**, see section 2.4.1.

The **Analyze Viral Hybrid Capture Panel Data (legacy)** template workflow is designed for detecting viruses, calculating abundances, and for calling variants for viruses identified from data generated using hybrid capture panels. The workflow performs read trimming, creates a QC report, cleans the dataset of host DNA, calculates viral abundances, and maps the reads to the most abundant viral reference for variant calling.

If your panel does not contain control genes, the workflow should be modified by right clicking on the workflow in the tool box and opening a copy of the workflow. Remove the Map Reads to Human Control Genes workflow element plus its input and output, and save the modified workflow. When you run the modified workflow, human control genes are no longer required.

Preliminary steps to run the Analyze Viral Hybrid Capture Panel Data (legacy) workflow

Before starting the workflow,

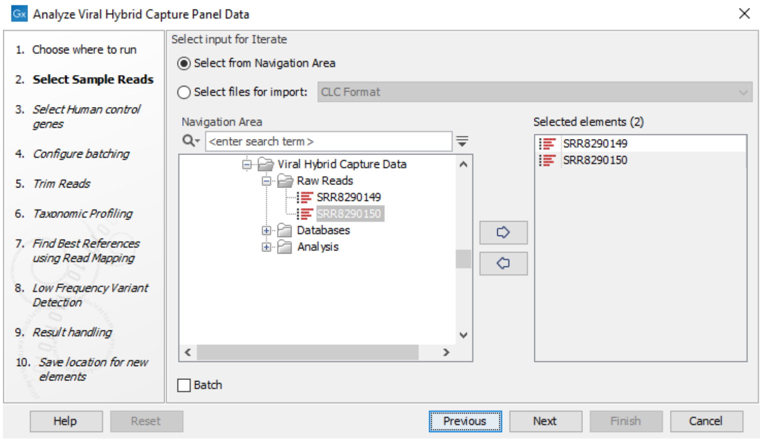
- Download reference viral genomes using either the **Download Custom Microbial Reference Database** (see section VI), the viral databases from the **Download Curated Microbial Reference Database** tool or create a database using the **Update Sequence Attributes in Lists** tool (https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Update_Sequence_Attributes_in_Lists.html)
- Create a taxonomic profiling index for calculating abundance (see section 15.4).

How to run the Analyze Viral Hybrid Capture Panel Data (legacy) template workflow

To run the workflow, go to:

Toolbox | **Legacy Tools** ( ) |**Analyze Viral Hybrid Capture Panel Data (legacy)** ( )

1. Specify the **sample(s)** or folder(s) of samples you would like analyze (figure 20.4) and click **Next**. Note that if you select several items, they will be run as batch units.



Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. **Select Sample Reads**
3. *Select Human control genes*
4. *Configure batching*
5. *Trim Reads*
6. *Taxonomic Profiling*
7. *Find Best References using Read Mapping*
8. *Low Frequency Variant Detection*
9. *Result handling*
10. *Save location for new elements*

Select input for Iterate

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area

<enter search term>

Viral Hybrid Capture Data
Raw Reads
SRR8290149
SRR8290150
Databases
Analysis

Selected elements (2)

SRR8290149
SRR8290150

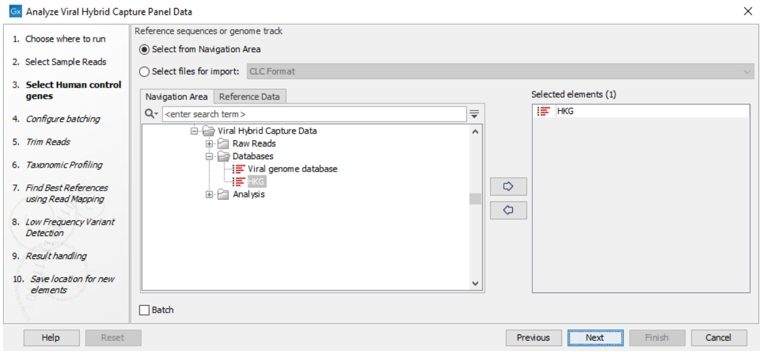
□ Batch

Figure 20.4: *Select the reads from the sample(s) you would like to analyze*

2. Specify the human control genes as a sequence list here (figure 20.5). Alternatively, if your panel does not contain control genes, the workflow should be modified by right clicking on the workflow in the tool box and opening a copy of the workflow. Then remove the **Map Reads to Reference** tool plus its input and output and save the modified workflow. When you run the modified workflow, human control genes are no longer required.

3. Define batch units using organisation of input data to create one run per input or use a metadata table to define batch units. Click **Next**.

4. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 20.6).



Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. **Select Human control genes**
4. *Configure batching*
5. *Trim Reads*
6. *Taxonomic Profiling*
7. *Find Best References using Read Mapping*
8. *Low Frequency Variant Detection*
9. *Result handling*
10. *Save location for new elements*

Reference sequences or genome track

● Select from Navigation Area

○ Select files for import: CLC Format

Navigation Area | Reference Data

<enter search term>

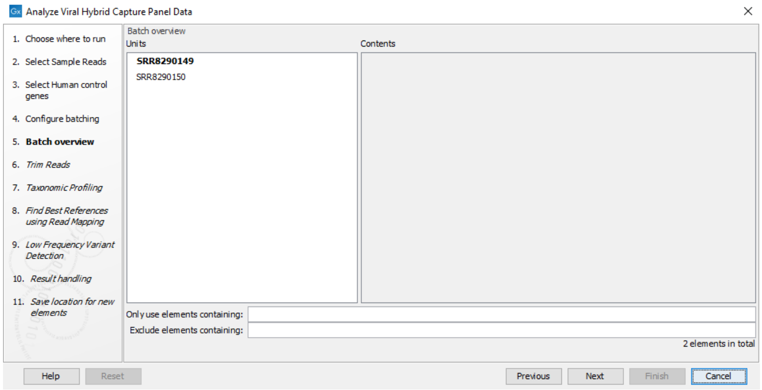
- Viral Hybrid Capture Data
  - Raw Reads
  - Databases
    - Viral genome database
    - HKG
  - Analysis

Selected elements (1)

HKG

□ Batch

Figure 20.5: *Select the human control genes or reference*



Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. Select Human control genes
4. Configure batching
5. **Batch overview**
6. *Trim Reads*
7. *Taxonomic Profiling*
8. *Find Best References using Read Mapping*
9. *Low Frequency Variant Detection*
10. *Result handling*
11. *Save location for new elements*

Batch overview

Units: **SRR8290149**, SRR8290150

Contents

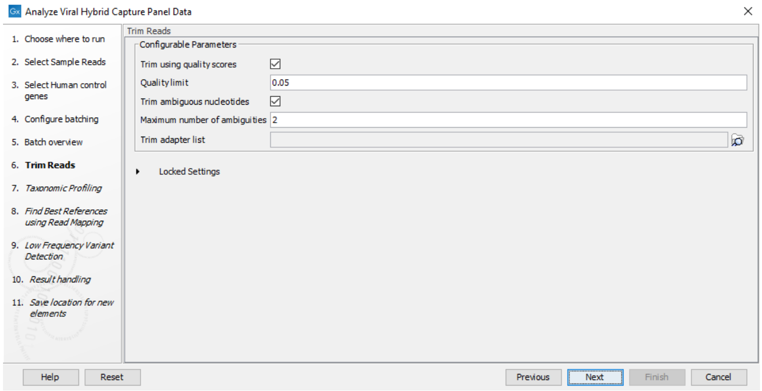
Only use elements containing:

Exclude elements containing:

2 elements in total

Figure 20.6: *Choose which of the samples present in the selected folder(s) you want to analyze.*

5. You can specify a **trim adapter list** and set up parameters if you would like to trim your sequences (figure 20.7).



Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. Select Human control genes
4. Configure batching
5. Batch overview
6. **Trim Reads**
7. *Taxonomic Profiling*
8. *Find Best References using Read Mapping*
9. *Low Frequency Variant Detection*
10. *Result handling*
11. *Save location for new elements*

Trim Reads

Configurable Parameters

Trim using quality scores ☑

Quality limit 0.05

Trim ambiguous nucleotides ☑

Maximum number of ambiguities 2

Trim adapter list

Locked Settings

Figure 20.7: *Choose trimming settings and optionally add an adapter trim list for trimming sequencing reads.*

The parameters that can be set are:

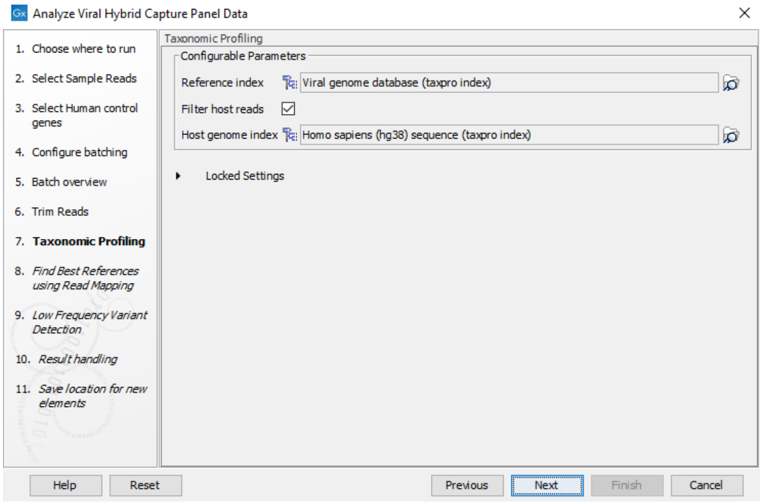
- **Trim ambiguous nucleotides**: if checked, this option trims the sequence ends based on the presence of ambiguous nucleotides (typically N).
- **Maximum number of ambiguities**: defines the maximal number of ambiguous nu-

cleotides allowed in the sequence after trimming.

- **Trim using quality scores**: if checked, and if the sequence files contain quality scores from a base-caller algorithm, this information can be used for trimming sequence ends.
- **Quality limit**: defines the minimal value of the Phred score for which bases will not be trimmed.
- **Trim adapter list**: Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your analysis (figure 20.7)

Learn about trim adapter lists at `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_adapter_list.html`)

6. In the next wizard window "Taxonomic Profiling", select the viral reference database index you will use to map the reads (figure 20.8). It is also possible to "Filter host reads". You must then specify the index of the host genome (in the case of human virus, the Homo sapiens GRCh38 for example). Note that if your panel uses human control genes, a taxonomic profiling index of the human genome should be used as "Host index".



Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. Select Human control genes
4. Configure batching
5. Batch overview
6. Trim Reads
7. **Taxonomic Profiling**
8. *Find Best References using Read Mapping*
9. *Low Frequency Variant Detection*
10. *Result handling*
11. *Save location for new elements*

Taxonomic Profiling

Configurable Parameters

Reference index: Viral genome database (taxpro index)

Filter host reads: ☑

Host genome index: Homo sapiens (hg38) sequence (taxpro index)

Locked Settings

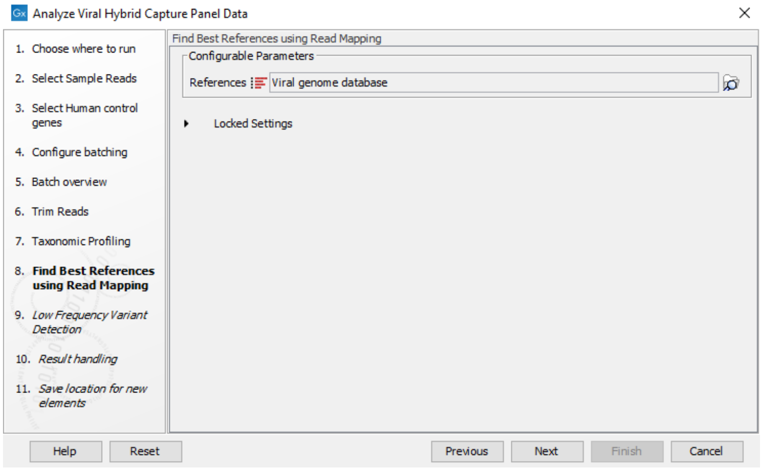
Figure 20.8: *Select taxonomic profiling index*

7. In the next wizard window, select the viral reference database you will use to find the best matching reference (figure 20.9). The best matching reference will be used for read mapping and variant calling. If you wish to have variant calls annotated with amino acid changes the input database should contain CDS annotations.

8. In the next wizard window, specify the parameters for the **Low Frequency Variant Detection** tool (figure 20.10). Note that variants are filtered after variant detection to coverage $>$ 30x and frequency $\geq$ 20%.

   The parameters that can be set are:

   - **Required significance**: The required significance level for low frequency variant calls.
   - **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.


Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. Select Human control genes
4. Configure batching
5. Batch overview
6. Trim Reads
7. Taxonomic Profiling
8. **Find Best References using Read Mapping**
9. *Low Frequency Variant Detection*
10. *Result handling*
11. *Save location for new elements*

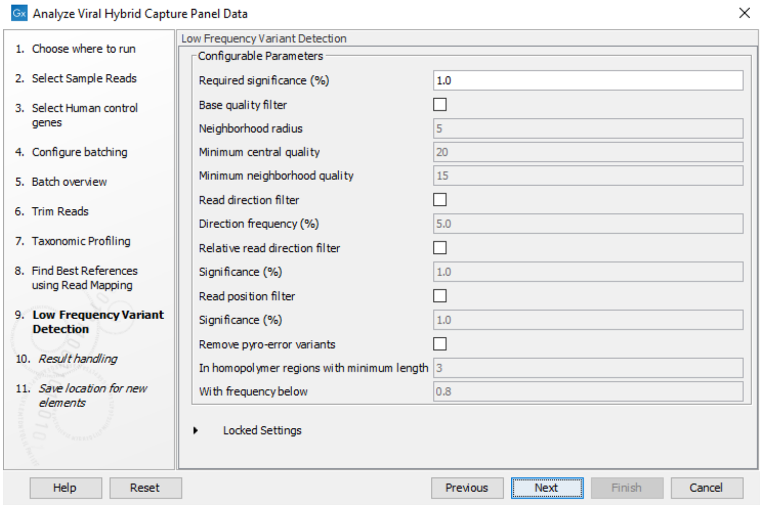
Find Best References using Read Mapping

Configurable Parameters

References: Viral genome database

Locked Settings

Figure 20.9: *Select viral reference database*


Analyze Viral Hybrid Capture Panel Data

1. Choose where to run
2. Select Sample Reads
3. Select Human control genes
4. Configure batching
5. Batch overview
6. Trim Reads
7. Taxonomic Profiling
8. Find Best References using Read Mapping
9. **Low Frequency Variant Detection**
10. *Result handling*
11. *Save location for new elements*

Low Frequency Variant Detection

Configurable Parameters

Required significance (%)	1.0
Base quality filter	□
Neighborhood radius	5
Minimum central quality	20
Minimum neighborhood quality	15
Read direction filter	□
Direction frequency (%)	5.0
Relative read direction filter	□
Significance (%)	1.0
Read position filter	□
Significance (%)	1.0
Remove pyro-error variants	□
In homopolymer regions with minimum length	3
With frequency below	0.8

Locked Settings

Figure 20.10: *Specify the parameters to be used by the Low Frequency Variant Detection tool.*

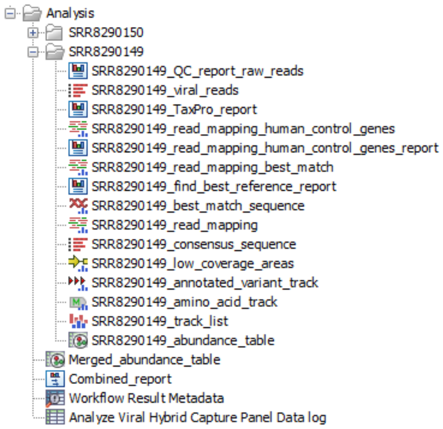
- **Neighborhood radius**: Determine how far away from the current variant the quality assessment should extend.
- **Minimum central quality**: Reads whose central base has a quality below the specified value will be ignored. This parameter does not apply to deletions since there is no "central base" in these cases.
- **Minimum neighborhood quality**: Reads for which the minimum quality of the bases is below the specified value will be ignored.
- **Read direction filter**: The read direction filter removes variants that are almost exclusively present in either forward or reverse reads.
- **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.

- **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**: This filter 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.
- **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 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.

9. In the Result handling window, pressing the button **Preview All Parameters** allows you to preview - but not change - all parameters. Choose to save the results (we recommend to create a new folder for it) and click **Finish**.

The output will be saved in the folder you selected. An example of the output can be seen in figure 20.11.



```
Analysis
  SRR8290150
  SRR8290149
    SRR8290149_QC_report_raw_reads
    SRR8290149_viral_reads
    SRR8290149_TaxPro_report
    SRR8290149_read_mapping_human_control_genes
    SRR8290149_read_mapping_human_control_genes_report
    SRR8290149_read_mapping_best_match
    SRR8290149_find_best_reference_report
    SRR8290149_best_match_sequence
    SRR8290149_read_mapping
    SRR8290149_consensus_sequence
    SRR8290149_low_coverage_areas
    SRR8290149_annotated_variant_track
    SRR8290149_amino_acid_track
    SRR8290149_track_list
    SRR8290149_abundance_table
  Merged_abundance_table
  Combined_report
  Workflow Result Metadata
  Analyze Viral Hybrid Capture Panel Data log
```

Figure 20.11: *Output of analysis of viral hybrid capture panel data*

The output generated for each sample is:

- **QC report raw reads**: QC report on the raw reads. Contains information on number of input reads, length, quality and nucleotide distributions.
- **Viral reads**: list of the sequences that were successfully trimmed and mapped to the best reference.
- **TaxPro report**: output from the **Taxonomic Profiling** tool. Contains information on number of reads mapping to the reference database and the host, if host filtering was enabled.

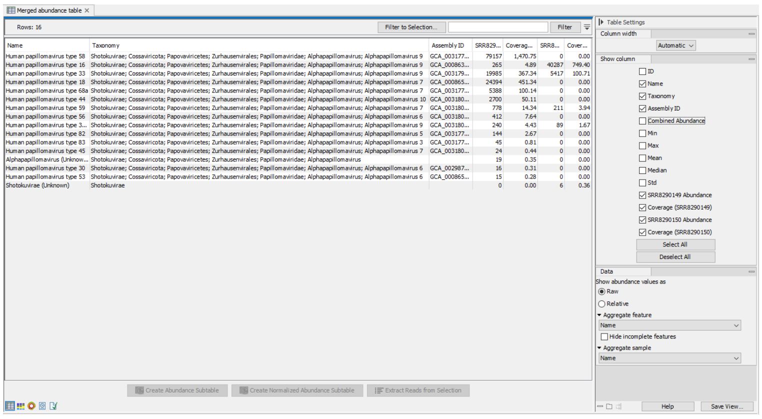
- **Read mapping human control genes**: mapping of the reads to the human control genes.
- **Read mapping human control genes report**: contains information on the read mapping to the selected controls such as number of reads mapped and read length distributions.
- **Best reference report**: contains information on "Best match" reference identified by **Find Best References using Read Mapping**, and the number of reads and unique reads mapped to this reference.
- **Best match sequence**: the "Best match" reference sequence as identified by the Find Best References using Read Mapping tool.
- **Read mapping**: reads mapped to the "Best match" viral reference. Output from **Local Realignment**.
- **Consensus sequence**: a consensus sequence generated from the read mapping. Consensus is not calculated in low coverage regions. These positions are instead replaced with Ns.
- **Low coverage areas**: track of regions of the best match reference genome with coverage < 30x.
- **Trim report**: report from the **Trim Sequences** tool (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html).
- **Abundance table**: contains abundance for all detected viral species.
- **Viral reads**: read mapping to the viral reference database.
- **Annotated variant track**: output from the **Low Frequency Variant Detection** tool. Note that variants are post filtered to coverage $>$ 30x and frequency $\geq$ 20%.
- **Amino acid track**: only generated if the best match contains CDS annotations.

For each batch analysis run, the following outputs are generated:

- **Combined report**: combines the information from the output report including QC, taxonomic profiling and mapping reports.
- **Merged abundance table**: An table containing abundance for all input samples. See figure 20.12 for an example of an analysis run of two HPV samples.

  Merged abundance tables can be used as input for various tools:

  - Alpha Diversity, see section 6.3
  - Beta Diversity, see section 6.4
  - Differential Abundance Analysis, see section 6.6
  - Create Heat Map for Abundance Table, see section 6.7



Rows: 16

Name	Taxonomy	Assembly ID	SRR829...	Coverag...	SRR8...	Cover...
Human papillomavirus type 58	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 9	GCA_003177...	79157	1,470.75	0	0.00
Human papillomavirus type 16	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 9	GCA_000863...	265	4.89	40287	749.40
Human papillomavirus type 33	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 9	GCA_003179...	19985	367.34	5417	100.71
Human papillomavirus type 18	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 7	GCA_000865...	24394	451.34	0	0.00
Human papillomavirus type 68a	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 7	GCA_003177...	5388	100.14	0	0.00
Human papillomavirus type 44	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 10	GCA_003180...	2700	50.11	0	0.00
Human papillomavirus type 59	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 7	GCA_003180...	778	14.34	211	3.94
Human papillomavirus type 56	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 6	GCA_003180...	412	7.64	0	0.00
Human papillomavirus type 3...	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 9	GCA_003180...	240	4.43	89	1.67
Human papillomavirus type 82	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 5	GCA_003177...	144	2.67	0	0.00
Human papillomavirus type 83	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 3	GCA_003177...	45	0.81	0	0.00
Human papillomavirus type 45	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 7	GCA_003180...	24	0.44	0	0.00
Alphapapillomavirus (Unknow...	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus		19	0.35	0	0.00
Human papillomavirus type 30	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 6	GCA_002987...	16	0.31	0	0.00
Human papillomavirus type 53	Shotokuvirae; Cossaviricota; Papovaviricetes; Zurhausenvirales; Papillomaviridae; Alphapapillomavirus; Alphapapillomavirus 6	GCA_000865...	15	0.28	0	0.00
Shotokuvirae (Unknown)	Shotokuvirae		0	0.00	6	0.36

Column width: Automatic

Show column
- ☐ ID
- ☑ Name
- ☑ Taxonomy
- ☑ Assembly ID
- ☐ Combined Abundance
- ☐ Min
- ☐ Max
- ☐ Mean
- ☐ Median
- ☐ Std
- ☑ SRR8290149 Abundance
- ☑ Coverage (SRR8290149)
- ☑ SRR8290150 Abundance
- ☑ Coverage (SRR8290150)

Data

Show abundance values as
- ● Raw
- ○ Relative

Aggregate feature: Name

☐ Hide incomplete features

Aggregate sample: Name

Figure 20.12: *Merged abundance table from analysis of two HPV samples*

Part X

Appendix

Chapter 21

Using the Assembly ID annotation

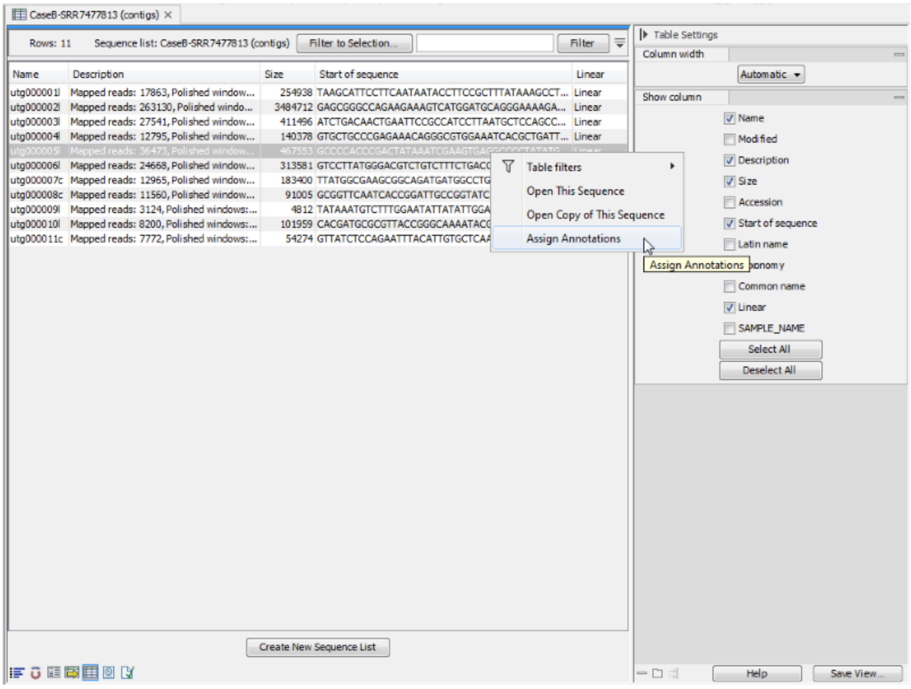
The **Assembly ID** annotation on sequences is used by many tools of the module to group sequences into meaningful entities, e.g. to group all contigs of a draft assembly. Tools that are aware of this annotation include

- Bin Pangenomes by Sequence, section 5.1.2
- Create K-mer Tree, section 9.2
- Create MLST Scheme, section 13.1
- Create Taxonomic Profiling Index, section 15.4
- Find Best Matches using K-mer Spectra, section 8.1
- Find Prokaryotic Genes, section 11.1

In order to see how these tools utilize the ''Assembly ID'' annotation, please read the tool documentation. In order to assign these annotations to sequences in a sequence list

1. Open the table view of a sequence list.
2. Select all rows corresponding to sequences that form a logical unit.
3. Right-click on the selection and choose **Assign annotations**, see figure 21.1.
4. Select **Assembly ID** from the dropdown menu in the Name field, see figure 21.2.
5. Enter a string in the Value field to uniquely identify the assembly.

For large sequence lists containing many assemblies it may be beneficial to use **Update Sequence Attributes in Lists** (see https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Update_Sequence_Attributes_in_Lists.html).



CaseB-SRR7477813 (contigs)

Rows: 11 Sequence list: CaseB-SRR7477813 (contigs)

Name	Description	Size	Start of sequence	Linear
utg000001l	Mapped reads: 17863, Polished window...	254938	TAAGCATTCCTTCAATAATACCTTCCGCTTTATAAAGCCT...	Linear
utg000002l	Mapped reads: 263130, Polished windo...	3484712	GAGCGGGCCAGAAGAAAGTCATGGATGCAGGGAAAAGA...	Linear
utg000003l	Mapped reads: 27541, Polished window...	411496	ATCTGACAACTGAATTCCGCCATCCTTAATGCTCCAGCC...	Linear
utg000004l	Mapped reads: 12795, Polished window...	140378	GTGCTGCCCGAGAAACAGGGCGTGGAAATCACGCTGATT...	Linear
utg000005l	Mapped reads: 36473, Polished window...	467553	GCCCCACCCGACTATAAATCGAAGTGAGGCCCCTATATG	Linear
utg000006l	Mapped reads: 24668, Polished window...	313581	GTCCTTATGGGACGTCTGTCTTTCTGACC	
utg000007c	Mapped reads: 12965, Polished window...	183400	TTATGGCGAAGCGGCAGATGATGGCCTG	
utg000008c	Mapped reads: 11560, Polished window...	91005	GCGGTTCAATCACCGGATTGCCGGTATC	
utg000009l	Mapped reads: 3124, Polished windows:...	4812	TATAAATGTCTTTGGAATATTATATTGGA	
utg000010l	Mapped reads: 8200, Polished windows:...	101959	CACGATGCGCGTTACCGGGCAAAATACG	
utg000011c	Mapped reads: 7772, Polished windows:...	54274	GTTATCTCCAGAATTTACATTGTGCTCAA	

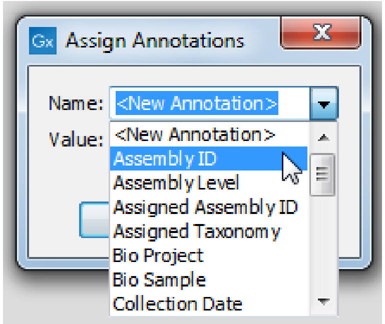
Table Settings

Column width: Automatic

Show column

☑ Name
☐ Modified
☑ Description
☑ Size
☐ Accession
☑ Start of sequence
☐ Latin name
xonomy
☐ Common name
☑ Linear
☐ SAMPLE_NAME

Figure 21.1: *Select the sequences forming a logical unit and right-click on the selection to assign annotations to these sequences.*



Assign Annotations

Name: <New Annotation>

Value:

Figure 21.2: *Select Assembly ID from the dropdown menu in the Name field and enter a string to uniquely identify the assembly in the Value field.*

Bibliography

[Anderson, 2001] Anderson, M. (2001). A new method for non-parametric multivariate analysis of variance. *Austral Ecology*, 26(1):32–46.

[Callahan et al., 2016] Callahan, B. J., McMurdie, P. J., Rosen, M. J., Han, A. W., Johnson, A. J., and Holmes, S. P. (2016). Dada2: High resolution sample inference from illumina amplicon data repository. *Nature Methods*, 13:581–583.

[Chen et al., 2012] Chen, J., Bittinger, K., Charlson, E. S., Hoffmann, C., Lewis, J., Wu, G. D., Collman, R. G., Bushman, F. D., and Li, H. (2012). Associating microbiome composition with environmental covariates using generalized unifrac distances. *Bioinformatics*, 28(16):2106–13.

[Curry et al., 2022] Curry, K. D., Wang, Q., Nute, M. G., Tyshaieva, A., Reeves, E., Soriano, S., Wu, Q., Graeber, E., Finzer, P., Mendling, W., et al. (2022). Emu: species-level microbial community profiling of full-length 16s rrna oxford nanopore sequencing data. *Nature methods*, 19(7):845–853.

[Douglas et al., 2020] Douglas, G. M., Maffei, V. J., Zaneveld, J. R., Yurgel, S. N., Brown, J. R., Taylor, C. M., Huttenhower, C., and Langille, M. G. I. (2020). PICRUSt2 for prediction of metagenome functions. *Nature Biotechnology*, 38(6):685–688.

[Dueholm et al., 2022] Dueholm, M. K. D., Nierychlo, M., Andersen, K. S., Rudkjøbing, V., Knutsson, S., Arriaga, S., Bakke, R., Boon, N., Bux, F., Christensson, M., Chua, A. S. M., Curtis, T. P., Cytryn, E., Erijman, L., Etchebehere, C., Fatta-Kassinos, D., Frigon, D., Garcia-Chaves, M. C., Gu, A. Z., Horn, H., Jenkins, D., Kreuzinger, N., Kumari, S., Lanham, A., Law, Y., Leiknes, T., Morgenroth, E., Muszyski, A., Petrovski, S., Pijuan, M., Pillai, S. B., Reis, M. A. M., Rong, Q., Rossetti, S., Seviour, R., Tooker, N., Vainio, P., van Loosdrecht, M., Vikraman, R., Wanner, J., Weissbrodt, D., Wen, X., Zhang, T., Nielsen, P. H., Albertsen, M., Nielsen, P. H., and Consortium, M. G. (2022). Midas 4: A global catalogue of full-length 16s rrna gene sequences and taxonomy for studies of bacterial communities in wastewater treatment plants. *Nature Communications*, 13(1):1908.

[Goodacre et al., 2018] Goodacre, N., Aljanahi, A., Nandakumar, S., Mikailov, M., and Khan, A. S. (2018). A reference viral database (rvdb) to enhance bioinformatics analysis of high-throughput sequencing for novel virus detection. *MSphere*, 3(2):e00069–18.

[Gupta et al., 2014] Gupta, S. K., Padmanabhan, B. R., Diene, S. M., Lopez-Rojas, R., Kempf, M., Landraud, L., and Rolain, J.-M. (2014). Arg-annot, a new bioinformatic tool to discover antibiotic resistance genes in bacterial genomes. *Antimicrobial agents and chemotherapy*, 58(1):212–220.

[Gurbich et al., 2023] Gurbich, T. A., Almeida, A., Beracochea, M., Burdett, T., Burgin, J., Cochrane, G., Raj, S., Richardson, L., Rogers, A. B., Sakharova, E., Salazar, G. A., and Finn, R. D. (2023). Mgnify genomes: A resource for biome-specific microbial genome catalogues. *Journal of Molecular Biology*, 435(14):168016. Computation Resources for Molecular Biology.

[Hasman et al., 2013] Hasman, H., Saputra, D., Sicheritz-Ponten, T., Lund, O., Svendsen, C. A., Frimodt-Møller, N., and Aarestrup, F. M. (2013). Rapid whole genome sequencing for the detection and characterization of microorganisms directly from clinical samples. *Journal of clinical microbiology*, pages JCM–02452.

[Kõljalg et al., 2020] Kõljalg, U., Nilsson, H. R., Schigel, D., Tedersoo, L., Larsson, K.-H., May, T. W., Taylor, A. F. S., Jeppesen, T. S., Frøslev, T. G., Lindahl, B. D., Põldmaa, K., Saar, I., Suija, A., Savchenko, A., Yatsiuk, I., Adojaan, K., Ivanov, F., Piirmann, T., Pöhönen, R., Zirk, A., and Abarenkov, K. (2020). The taxon hypothesis paradigm - on the unambiguous detection and communication of taxa. *Microorganisms*, 8(12).

[Kaas et al., 2014] Kaas, R. S., Leekitcharoenphon, P., Aarestrup, F. M., and Lund, O. (2014). Solving the problem of comparing whole bacterial genomes across different sequencing platforms. *PLOS ONE*.

[Kaminski et al., 2015] Kaminski, J., Gibson, M. K., Franzosa, E. A., Segata, N., Dantas, G., and Huttenhower, C. (2015). High-specificity targeted functional profiling in microbial communities with shortbred. *PLoS Comput. Biol.*

[Kang et al., 2015] Kang, D., Froula, J., Egan, R., and Wang, Z. (2015). Metabat, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ*, 3:e1165.

[Kelley and Salzberg, 2010] Kelley, D. and Salzberg, S. (2010). Clustering metagenomic sequences with interpolated markov models. *BMC Bioinformatics*, 11:544.

[Larsen et al., 2014] Larsen, M. V., Cosentino, S., Lukjancenko, O., Saputra, D., Rasmussen, S., Hasman, H., Sicheritz-Pontén, T., Aarestrup, F. M., Ussery, D. W., and Lund, O. (2014). Benchmarking of methods for genomic taxonomy. *Journal of clinical microbiology*, 52(5):1529--1539.

[McDonald et al., 2022] McDonald, D., Jiang, Y., Balaban, M., Cantrell, K., Zhu, Q., Gonzalez, A., Morton, J. T., Nicolaou, G., Parks, D. H., Karst, S., et al. (2022). Greengenes2 enables a shared data universe for microbiome studies. *bioRxiv*, pages 2022–12.

[Narayan et al., 2020] Narayan, N. R., Weinmaier, T., Laserna-Mendieta, E. J., Claesson, M. J., Shanahan, F., Dabbagh, K., Iwai, S., and DeSantis, T. Z. (2020). Piphillin predicts metagenomic composition and dynamics from DADA2-corrected 16S rDNA sequences. *BMC Genomics*, 21(1):56.

[Quast et al., 2012] Quast, C., Pruesse, E., Yilmaz, P., Gerken, J., Schweer, T., Yarza, P., Peplies, J., and Glöckner, F. O. (2012). The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Research*, 41(D1):D590–D596.

[Sedlar et al., 2017] Sedlar, K., Kupkova, K., and Provaznik, I. (2017). Bioinformatics strategies for taxonomy independent binning and visualization of sequences in shotgun metagenomics. *Computational Structural Biotechnology Journal*, 15:48–55.

[Ye and Doak, 2009] Ye, Y. and Doak, T. G. (2009). A Parsimony Approach to Biological Pathway Reconstruction/Inference for Genomes and Metagenomes. *PLoS Computational Biology*, 5(8):e1000465.

[Zankari et al., 2017] Zankari, E., Allesï¿½e, R., Joensen, K. G., Cavaco, L. M., Lund, O., and Aarestrup, F. M. (2017). Pointfinder: a novel web tool for wgs-based detection of antimicrobial resistance associated with chromosomal point mutations in bacterial pathogens. *Journal of Antimicrobial Chemotherapy*, 72(10):2764–68. https://doi.org/10.1093/jac/dkx217.