User manual for

**CLC Microbial Genomics Module 3.6.1**

Windows, macOS and Linux

October 5, 2018

This software is for research purposes only.
5.4 Align OTUs with MUSCLE .................................................. 33
5.5 Amplicon-Based Analysis Workflows .............................. 34
   5.5.1 Data QC and OTU Clustering workflow .................. 34
   5.5.2 Estimate Alpha and Beta Diversities workflow ....... 35

6 Taxonomic Analysis ......................................................... 37
   6.1 Taxonomic Profiling .................................................. 37
      6.1.1 Taxonomic profiling abundance table ................. 39
      6.1.2 Taxonomic Profiling report .......................... 42
      6.1.3 Sorted Reads Files .................................. 43
   6.2 Workflows ........................................................ 44
      6.2.1 Data QC and Clean Host DNA ......................... 44
      6.2.2 Data QC and Taxonomic Profiling .................. 45
      6.2.3 Merge and Estimate Alpha and Beta diversities .... 47

7 Functional analysis ...................................................... 48
   7.1 Find Prokaryotic Genes (beta) ................................. 49
   7.2 Annotate CDS with Best BLAST Hit ......................... 49
   7.3 Annotate CDS with Best DIAMOND Hit ................... 51
   7.4 Annotate CDS with Pfam Domains ............................ 51
   7.5 Build Functional Profile ..................................... 53
      7.5.1 Functional profile abundance table ............... 54

8 Abundance analysis ........................................................ 58
   8.1 Merge Abundance Tables ....................................... 58
   8.2 Alpha Diversity .................................................. 58
      8.2.1 Alpha diversity measures ............................ 60
   8.3 Beta Diversity .................................................. 61
      8.3.1 Beta diversity measures ............................... 63
   8.4 PERMANOVA Analysis .......................................... 63
   8.5 Differential Abundance Analysis ............................ 65
   8.6 Create Heat Map for Abundance Table ..................... 66
   8.7 Add Metadata to Abundance Table ........................... 69
CONTENTS

8.8 Convert Abundance Table to Experiment ................................. 70

III Typing and Epidemiology (beta) ................................. 72

9 Introduction to Typing and Epidemiology (beta) .................. 73

10 Handling of metadata and analysis results ......................... 75
   10.1 Importing Metadata Tables ........................................ 75
   10.2 Associating data elements with metadata ...................... 77
   10.3 Create a Result Metadata Table ................................. 78
   10.4 Running an analysis directly from a Result Metadata Table ...... 80
      10.4.1 Filtering in Result Metadata Table ..................... 81
      10.4.2 Filtering in a SNP-Tree creation scenario .............. 82
   10.5 Add to Result Metadata Table .................................... 82
   10.6 Use Genome as Result .............................................. 84

11 Workflow templates ......................................................... 86
   11.1 Map to Specified Reference ....................................... 86
      11.1.1 How to run the Map to Specified Reference workflow on a single sample: 89
      11.1.2 How to run the Map to Specified Reference workflow on a batch of samples: 93
   11.2 Type Among Multiple Species ..................................... 95
      11.2.1 Preliminary steps to run the Type Multiple Species workflow .......... 95
      11.2.2 How to run the Type among Multiple Species workflow for a single sample 96
      11.2.3 Example of results obtained using the Type among Multiple Species workflow 102
      11.2.4 How to run the Type among Multiple Species workflow on a batch of samples: 103
   11.3 Type a Known Species ................................................. 105
      11.3.1 Preliminary steps to run the Type a Known Species workflow .......... 106
      11.3.2 How to run the Type a Known Species workflow for a single sample .... 106
      11.3.3 Example of results obtained using the Type a Known Species workflow ... 112
      11.3.4 How to run the Type a Known Species workflow on a batch of samples: ... 112
   11.4 Extract Regions from Tracks ...................................... 115

12 Find the best matching reference ....................................... 117
   12.1 Find Best Matches using K-mer Spectra .......................... 117
12.2 From samples best matches to a common reference for all ............................ 120

13 Resistance typing ......................................................... 122
   13.1 Find Resistance .................................................. 122

14 Phylogenetic trees using SNPs or k-mers .......................................... 124
   14.1 Create SNP Tree ................................................... 124
      14.1.1 SNP tree output report ..................................... 128
      14.1.2 Visualization of SNP Tree including metadata and analysis result metadata 130
      14.1.3 SNP Tree Variants editor viewer .............................. 130
      14.1.4 SNP Matrix ...................................................... 131
   14.2 Create K-mer Tree .................................................. 132
      14.2.1 Visualization of K-mer Tree for identification of common reference ............. 133

15 NGS MLST ................................................................. 136
   15.1 Schemes visualization and management ........................................ 137
   15.2 Identify MLST ....................................................... 138
   15.3 Identify MLST Scheme from Genomes ....................................... 140

IV Databases .................................................................. 141

16 Databases for Taxonomic Profiling .................................................. 142
   16.1 Create Microbial Reference Database ................................... 142
   16.2 Set Up Microbial Reference Database .................................... 144
   16.3 Download Pathogen Reference Database ...................................... 147
      16.3.1 Extracting a subset of a database .................................. 148
      16.3.2 Download Pathogen Reference Database output report ....................... 149

17 Databases for Functional Analysis .................................................. 151
   17.1 Download Database for Find Resistance ..................................... 151
   17.2 Set Up Gene Database .................................................. 151
   17.3 Download Protein Database .............................................. 153
   17.4 Download GO Database ............................................... 154

18 Databases for Amplicon-Based Analysis ............................................. 155
18.1 Download Amplicon-Based Reference Database .................................................. 155
18.2 Set Up Amplicon-Based Reference Database ..................................................... 156

19 Databases for NGS-MLST .................................................................................. 157
19.1 Download MLST Schemes ................................................................................. 157
19.2 Download other MLST Schemes ...................................................................... 158
19.3 Create MLST Schemes ..................................................................................... 158
19.4 Add NGS MLST Report to Scheme .................................................................. 159
19.5 Merge MLST Schemes ...................................................................................... 160
19.6 Add Sequences to MLST Schemes .................................................................. 161

V Panel Support ....................................................................................................... 162

20 QIAseq 16S/ITS Demultiplexer ........................................................................... 163

VI Appendices ......................................................................................................... 166

21 Legacy tools ......................................................................................................... 167
21.1 Optional Merge Paired Reads .......................................................................... 167
21.2 Fixed Length Trimming ..................................................................................... 169

22 Licensing requirements for the CLC Microbial Genomics Module .................... 170
22.1 Workbench licenses ........................................................................................ 170
  22.1.1 Request an evaluation license ...................................................................... 171
    Direct download .................................................................................................. 171
    Go to license download web page ..................................................................... 171
  22.1.2 Download a license using a license order ID .............................................. 172
    Direct download .................................................................................................. 173
    Go to license download web page ..................................................................... 173
  22.1.3 Import a license from a file .......................................................................... 174
  22.1.4 Configure license server connection ............................................................ 175
    Borrowing a license ........................................................................................... 176
    Common issues when using a network license .................................................. 177
Part I

Introduction
Chapter 1

Introduction to CLC Microbial Genomics Module

Welcome to CLC Microbial Genomics Module 3.6.1 – 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.

### 1.2 Contact information

CLC Microbial Genomics Module is developed by:

QIAGEN Aarhus  
Silkeborgvej 2  
Prismet  
8000 Aarhus C  
Denmark

http://www.qiagenbioinformatics.com  
Email: ts-bioinformatics@qiagen.com

The QIAGEN Aarhus team is continuously improving the CLC Workbench with our users’ interests in mind. We welcome all requests and feedback from users, as well as suggestions for new features or more general improvements to the program. If you have questions or comments regarding the program, please contact the support team as described here: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Getting_help.html.

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

- manuals (http://www.qiagenbioinformatics.com/support/manuals/);
- tutorials (http://www.qiagenbioinformatics.com/support/tutorials/);
Chapter 2

System requirements and installation

2.1 System requirements

To work with CLC Microbial Genomics Module 3.6.1 you will need to have CLC Genomics Workbench version 11.0 or higher or Biomedical Genomics Workbench 5.0 or higher installed on your computer. With exception of the two editors below, the system requirements of CLC Microbial Genomics Module 3.6.1 are the same as the ones required for the CLC Genomics Workbench:

- OS X 10.10, 10.11 and macOS 10.12, 10.13
- Linux: RHEL 6.7 and later, SUSE Linux Enterprise Server 11 and later. The software is expected to run without problem on other recent Linux systems, but we do not guarantee this.
- 2 GB RAM required
- 8 GB RAM recommended
- 1024 x 768 display required
- 1600 x 1200 display recommended
- Intel or AMD CPU required

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

Special requirements for De Novo Assemble Metagenome. It is recommended to have at least 16 GB RAM when running the De Novo Assemble Metagenome.
The **PCoA 3D viewer** requirements are the same as the 3D Molecule Viewer:

- **System requirements**
  - A graphics card capable of supporting OpenGL 2.0.
  - Updated graphics drivers. Please make sure the latest driver for the graphics card is installed.
  - Indirect rendering (such as x11 forwarding through ssh), remote desktop connection/VNC, and running in virtual machines are not supported.

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

The **Sunburst viewer** makes use of JavaFX and may not work on older Linux kernels. An updated list of requirements for JavaFX can be found at [http://www.oracle.com/technetwork/java/javafx/downloads/supportedconfigurations-1506746.html](http://www.oracle.com/technetwork/java/javafx/downloads/supportedconfigurations-1506746.html).

## 2.2 Installation of modules

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

Modules are installed and uninstalled using the plugin manager.

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

From within the Plugins manager (see figure 2.2), choose the Download Plugins tab and click on CLC Microbial Genomics Module. Then click on the button labeled **Download and Install**.

**Accepting the license agreement**

Part of the installation involves checking and accepting the end user license agreement (EULA) as seen in figure 2.1.

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

If you are working on a system not connected to the internet, then you can also install the plugin by downloading the *.cpa file from the plugins page of our website


Then click on the button at the bottom of the Plugins manager labeled **Install from File**.

You need to restart the Workbench before the module is ready for use.
Figure 2.1: Read the license agreement carefully.

Figure 2.2: The plugins that are available for download are listed in the Download Plugins tab of the Plugin Manager.

2.3 Plugins licenses

When you have installed the CLC Microbial Genomics Module and start a tool from that plugin for the first time, you will meet the license assistant shown in figure 2.3. You can also manually start up the License Manager for a Workbench plugin: first open the Plugin Manager (see Section 2.2), select the relevant plugin or module, and press the button labeled Import a new license.

To install a license, you must be running the program in administrative mode. On Linux and Mac, this means you must be logged in as an administrator. On Windows, you can right-click the program shortcut and choose “Run as Administrator”.

The following options are available. They are described in detail in the (appendices 22.1).

- **Request an evaluation license.** Request a fully functional, time-limited license.
You need a license…

In order to load the plugin “CLC Gene Expression Module” you need a valid license. Please choose how you would like to obtain a license for the 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 Server connection**
  Configure the necessary connection for the software to connect to a CLC License Server that hosts network-licensed products. This option also allows you to alter or disable an existing configuration.

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

To use the Download option in the License Manager, your machine must be able to access the external network. If this is not the case, please see section 22.2.5.

### 2.4 Uninstalling modules

Workbench modules are uninstalled using the Plugin Manager:

**Help in the Menu Bar | Plugins… (_plugins) or Plugins (plugins) in the Toolbar**

This will open the dialog shown in figure 2.4.

Figure 2.4: The plugin manager with plugins installed.
To uninstall the module, select the entry for the CLC Microbial Genomics Module and click on the **Uninstall** button.

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

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

### 2.5 Server License

If you wish to use the tools and functionalities of CLC Microbial Genomics Module with CLC Genomics Server, you must purchase CLC Microbial Genomics extension license and install it on your CLC Server as explained in the following steps:

1. Install the **license(s)** to each machine with CLC Server software installed, as described in the appendix (see section 22.2).


3. Install **the server module** on the server. Note that in a server setup, the plugin does not need to be manually installed on any machine acting as an execution node but **only on the master CLC Server**.


   ![Figure 2.5: Installing and uninstalled CLC Server plugins is done via the Plugins section of the web administrative interface.](image)

5. Click on the **Browse** button and locate the *.cpa file for the plugin to install.

7. **For job node setups only**: Wait until 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.

8. In the web administrative interface on the master CLC Server, check that the plugin is enabled for each job node. This is described in more detail in the CLC Server manual at: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsserver/current/admin/index.php?manual=Configuring_your_setup.html.

To uninstall a CLC Server plugin, simply click on the button that has Uninstall on its label next to the relevant plugin.
Part II

Metagenomics
Chapter 3

Introduction to Metagenomics

Amplicon-based microbiome analysis takes advantage of DNA molecular techniques and sequencing technology in order to comprehensively retrieve specific regions of microbial genomic DNA useful for taxonomic identification. In a classic microbiome analysis workflow, total genomic DNA is extracted from the sample(s) of interest and a chosen amplicon (often the small-subunit ribosomal RNA 16S locus, or the fungal Internal Transcribed Spacer (ITS) region) is PCR amplified and sequenced using an NGS machine. The bioinformatics task is then to assign taxonomy to the reads and tally their occurrences. The tools from Amplicon-Based Analysis are designed to cluster all reads within a certain percentage of similarity into Operational Taxonomic Units (OTUs) where they are then represented by a single sequence.

Although 16S and ITS are both a taxonomically and phylogenetically informative marker, the resolution of these studies is limited. So rather than focusing on a single locus, the shotgun genomic sequencing of entire communities has become a viable alternative thanks to the decreasing costs of sequencing protocols. This approach is applicable to samples of uncultured microbiota and avoids some of the limitations of amplicon sequencing. The Taxonomic Analysis tools of Microbial Genomics Module are designed to determine which known organisms are in a sample, and how abundant they are by mapping each input reads to a reference database of complete genomes - as opposed to amplicon-based OTU databases.

It is also possible to annotate a whole metagenome shotgun sequencing dataset with BLAST hits or Pfam protein families and GO terms using the tools from the Functional Analysis folder and the third-party MetaGeneMark plugin. While GO is a hierarchy of higher-level functional catagories, Pfam (Protein families) classifies proteins into families of related proteins with similar function, allowing to build the functional profile of a microbial community.

All abundance tables generated by the methods above can be visualized in stacked bar charts, stacked area charts, sunburst charts and heat maps. In addition, the tools included in the Abundance Analysis folder will perform various statistical analyses, highlighting the results of the metagenomics study performed. The reference databases needed for clustering, profiling and annotating can be downloaded and restructured using the tools from the Databases folder of the Toolbox (see IV).

Please Note that the functionality of the Functional analysis folder described within this section is in beta. As this is still a very active research area, the software is accordingly also under active development and subject to change without notice.
Chapter 4

De Novo Assemble Metagenome

Adapters should be removed from sequences before assembly using the Trim Reads tool. 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 assembling is not generally necessary as the assembler itself should weed out or correct bad quality regions. However, trimming off 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 assemble a metagenome de novo, run the tool:

Metagenomics | De Novo Assemble Metagenome

Select one or more sequence lists or single sequences to assemble, then set the parameters for the assembly. This will show a dialog similar to the one in figure 4.1.

![Figure 4.1: Setting parameters for the assembly.](image)

The first parameter allows you to specify the **Minimum contig length**. Contigs below this length (default value 200 bp) will not be reported. The assembler will often produce shorter contigs for very complex datasets containing reads from many closely related species. In such a case, it is often wise to set a lower threshold in order to cover a larger proportion of the metagenome with contigs. Similarly, for metagenomes of low complexity, it is often wise to set a higher threshold in order to avoid duplication.
After setting the Minimum contig length, you need to choose between running the assembler in Fast mode or Longer contigs mode. In Fast mode, the assembler is iterated once with a predefined wordsize ($k = 21$). In Longer contigs mode, 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 naturally requires up to three times more compute time.

At the bottom, there is an option to Perform scaffolding. When this option is selected, the assembler will attempt to join contigs using paired-end information as the last step of the assembly process. Since paired-end information is needed to perform scaffolding, this option is disabled if the input dataset does not contain any paired-end reads.

The result of the de novo assembly is a list of contigs in a sequence list. If the Perform scaffolding option was selected in the previous wizard step, the resulting scaffolds will appear at the end of the sequence list and will be named scaffold_1, scaffold_2, etc. If Create report was selected, the tool will create a summary report containing basic statistics on both the input reads and the output contigs.
Chapter 5

Amplicon-Based Analysis

In the Amplicon-Based Analysis folder you will find all the tools needed to cluster reads at some level of similarity into pseudo-species called Operational Taxonomic Units (OTUs), where all reads within for example 97% similarity are clustered together and represented by a single sequence. To this effect, we recommend the use of workflows, where all the steps needed to cluster the reads in OTUs have been pre-organized and optimized to facilitate your analysis. These workflows are found in the Workflows folder inside the Amplicon-based OTU Clustering folder (see section 5.5).

5.1 Filter Samples Based on the 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 the Number of Reads tool.

To run the tool, go to

Microbial Genomics Module | Metagenomics | Amplicon-Based Analysis | Filter Samples Based on the 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 5.1).

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 because it does not contain the reads of poor quality that failed the Remove the samples with Low Coverage analysis.

5.2 OTU clustering

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

To run the tool, go to 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 merge them into pairs, and align the resulting paired reads to OTUs. Reads that cannot be merged will be independently aligned to reference OTUs. Then both reads of a pair will be assigned to the 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 ACGACGACGnnnnnnnnnnnnnnnTACTACTAC.

If a read cannot be put into an already existing OTU (because there is no single OTU that is similar enough, i.e., within 97% similarity), the algorithm tries to optimize the alignment score by allowing to "cross over" 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 some "promising candidates" found via a k-mer search. If the best match that can be constructed has at least one crossover and the "constructed alignment" is at least as good as the "similarity percentage", then the read is being 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.

The OTU clustering tool produces several outputs:
• a sequence list of the OTU centroids
• abundance tables with the newly created OTUs and the chimeras. Each table give abundance of the OTU or chimeras at each site, as well as the total abundance for all samples.
• a report that summarizes the results of the OTU clustering
• if the input data is paired-end, a report about the merging of overlapping reads

5.2.1 OTU clustering parameters

After having selected the sequences you would like to cluster, the wizard offers to set some general parameters (see figure 5.2).

![Figure 5.2: Settings for the OTU clustering tool.](image)

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

The following parameters can then be set:

• **OTU database** Specify here the reference database to be used for Reference based OTU clustering. Reference databases can be created by the Download Amplicon-Based Reference Database or the Set Up Amplicon-Based Reference Database tools.

• **Similarity percent specified by OTU database** Allows to use the same similarity percent value (see below) that 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.

• **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 duplicates for specific read-data before it will be included in further analyses. For instance, if set to 2, at least two reads with the same exact nucleotides needs to exist in the input for the data to propagate to further analysis. Other data will be thrown away. This can for instance be used to filter out singletons. **Note** that matches does not need to be exact when the **Fuzzy match duplicates** option is used.

- **Fuzzy match duplicates**: Specifies how duplicates are defined. If the option is not selected two reads are only duplicates if they are exactly equal. If the option is selected, two reads are duplicates if they are almost equal, i.e. all differences are SNVs and there are not too many of them (≤ 2%). This pseudo-merging is done by lexicographically sorting the input and looking in the neighborhood of the read being processed. The reads are processed from most abundant (in a completely equivalent sense) to the least. In this way two singletons can for instance be pseudo-merged together and be included for further study despite the **Minimum occurrences** option having specified 2. Upon further analysis a group can be split into several OTUs if not all members are within the specified threshold from the "OTU-leader".

- **Find best match**: If the option is not selected, the 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 references while allowing "cross overs". 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 5.3.

![Alignment parameters](image)

**Figure 5.3: 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.

5.2.2 OTU clustering tool outputs

The OTU clustering tool produces several outputs: a sequence list of the OTU centroids, and two abundance tables with the newly created OTUs or the chimeras. Each table give abundance of the OTU or chimeras at each site as well as the total abundance for all samples. Note that if the input contains paired-end sequences, each pair is counted as one read.

In the OTU merge report - generated if the input reads were paired - the following statistics are given for each sequence list input in the tool: a summary of merged and non merged sequences, an alignment score distribution and a merged pairs length distribution.

The tool also produces a report (figure 5.4).

<table>
<thead>
<tr>
<th>Overall summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input database size</td>
</tr>
<tr>
<td>99,322</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Read summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of reads</td>
</tr>
<tr>
<td>82,371</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sample details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample</td>
</tr>
<tr>
<td>CT-A-B.1001_R1_001 (paired) trimmed (paired) merged fixedLength coverage OK</td>
</tr>
<tr>
<td>CT-A-C.1001_R1_001 (paired) trimmed (paired) merged fixedLength coverage OK</td>
</tr>
<tr>
<td>CT-B-A.1001_R1_001 (paired) trimmed (paired) merged fixedLength coverage OK</td>
</tr>
<tr>
<td>CT-B-B.1001_R1_001 (paired) trimmed (paired) merged fixedLength coverage OK</td>
</tr>
</tbody>
</table>

Figure 5.4: Example of report produced by the OTU clustering tool.

The content of the report is stated below.

- In the section Overall summary
– **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.

• In the section **Read summary**
  – **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.

• In the section **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.

### 5.2.3 Visualization of OTU abundance tables

The OTU abundance tables containing the newly created OTUs or the chimeras give abundance of the OTU or chimeras at each site as well as the total abundance for all samples. There are a number of ways of visualizing the contents of an OTU abundance table:

• **Table view** (figure 5.5)
  The table display 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.
CHAPTER 5. AMPLICON-BASED ANALYSIS

Figure 5.5: OTU abundance table.

- **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 a random of numbers and letters. If the same name is present more than once, then the OTUs will have a trailing number "-00123" like readName-12345.
- and if there is no name (for new clusters where reads have no name), something like OTU-12345 is assigned.

This will occur when one choose 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 8.7), 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. 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 table, than the merge should be redone excluding the samples with zero control read counts.

• Stacked Bar Chart and Stacked Area Chart (figure 5.6) In the Stacked Bar (figure 5.6) and Stacked Area Charts (figure 5.7), 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 be specified at the same taxonomy level as the one use 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
CHAPTER 5. AMPLICON-BASED ANALYSIS

Figure 5.7: Stacked area of the microbial community at the phylum level for 12 different sites. 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 5.8).

Figure 5.8: Sunburst view of the microbial community showing all taxa belonging to the kingdom bacteria.

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 5.9).
5.2.4 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 $\rightarrow$ Import ($\rightarrow$ Standard Import... ($\rightarrow$) 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.

- **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.

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.
5.3 Remove OTUs with Low Abundance

Low abundance OTUs can be 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

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.

5.4 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:

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 5.10).

Figure 5.10: 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 phylogenetic 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 [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Maximum_Likelihood_Phylogeny.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Maximum_Likelihood_Phylogeny.html)

### 5.5 Amplicon-Based Analysis Workflows

A workflow is a series of tools connected to one another to be processed in sequential order enabling repeated executions. They require that you provide the necessary input files and edit the parameter settings, and the workflow will output all relevant results.

The Amplicon-Based Analysis folder contains two workflows that you can start here:

Microbial Genomics Module (⊂) | Metagenomics (⊂) | Amplicon-Based Analysis (⊂) | Workflows (⊂) | Data QC and OTU Clustering (⊂) or Estimate Alpha and Beta Diversities (⊂)

To explore a workflow and see the tools it is made of, select the workflow and right click on its name to select the **Open Copy of Workflow** option.

#### 5.5.1 Data QC and OTU Clustering workflow

The **Data QC and OTU Clustering** workflow consists of 3 tools being executed sequentially (figure 5.11). 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 the 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.
5.5.2 Estimate Alpha and Beta Diversities workflow

The Estimate Alpha and Beta Diversities workflow consists of 5 tools and requires only the OTU table as input file (figure 5.12).

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 http://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
Figure 5.12: Layout of the Alpha and Beta Diversities workflow.

- **Beta diversity** tool

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.
Chapter 6

Taxonomic Analysis

6.1 Taxonomic Profiling

The Taxonomic Profiling tool is designed to determine which known organisms are in a whole shotgun metagenomic sample, and how abundant they are. To this end, the tool will map each input read to a reference database of complete genomes. If a host organism genome is provided, the mapping phase will disregard reads that it deems to have originated from the host. Paired reads that cannot map as an intact pair will also be dismissed. After mapping, the tool performs qualification (by assigning the read to a taxon in the database if a match is found) and quantification of the abundance of each qualified taxon, and finally compile the results in an abundance table.

The purpose of the qualification phase is to determine whether a particular taxon is represented in a sample. The qualification is based on the confidence score that a reference did not receive its reads by pure chance. Any taxon with a confidence score < 0.995 will be ignored and the reads assigned to it will be reassigned to its closest qualified ancestor.

The purpose of the quantification phase is to estimate how abundant the qualified taxons are. It is based on the number of reads assigned to that taxon.

The detection limit of the tool is now controlled by the single read mapping parameter "Minimum seed length". Increasing this value will give higher precision in the taxa called (true positives), while lowering it will give more taxa called at the cost of precision (more false positives).

To run the Taxonomic Profiling tool, go to Microbial Genomics Module | Metagenomics | Taxonomic Analysis | Taxonomic Profiling.

You can select only one read file to analyze (figure 6.1). If the sample to be analyzed is split up into several files, the files need to be merged with the Create Sequence List tool before running the Taxonomic Profiling tool.

In the "Parameters" dialog, provide the reference database you will use to map the reads (figure 6.2). Note that the first time you run the tool with a given database, the analysis will take longer because it is indexing and caching the database as indicated by the warning message in the dialog. Analysis time will be improved in subsequent runs, when the workbench is able to use the index generated the first time around.

In that dialog, you can choose to "Filter host reads". You must then specify the host genome (for
example the Homo sapiens hg38 in the case of human microbiota).

The read-mapping parameters used in the taxonomic profiler are the standard read-mapping parameters (see [link](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters.html)) except for the Minimum seed length, which may be specified explicitly here. The Minimum seed length defines the minimum seed (word) size, i.e., perfect match-length, for a position in the reference to be considered a valid candidate when matching the read. If no seed longer than this length can be found in the database, the read is considered "unmapped". Increasing the Minimum seed length will give higher precision in the results, while lowering it will give a higher recall rate but with more possible false positives.

Finally, checking the option "Auto-detect paired distances" will generate an estimate of the paired distance range in an additional section of the report output by the tool.

In the last dialog, choose from the different output options and Open or Save the results (figure 6.3).

The tool will generate by default an abundance table as well as a report with a list of the taxons and their abundances. You can choose to output additional files such as a sequence lists of the reads matching the reference database and those matching the host, as well as the unclassified reads.
6.1.1 Taxonomic profiling abundance table

The Taxonomic profiling abundance table displays the names of the identified taxa (assemblies), along with their full taxonomy, a coverage estimate, the total amount of reads found in the sample that are associated with this taxon and the confidence score for the taxonomic assignment. The table can be visualized using the Stacked bar charts and Stacked area charts function, as well as the Sunburst charts.

- **Table view** (figure 6.4)

The table displays the following columns:

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

– **Assembly ID**: the id of the assembly (typically genbank assembly accession numbers), as specified by the reference database.

– **Combined Abundance**: total number of reads for the taxa 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**: (for example LC1 in the table above): number of reads for each sample (calculated during the quantification phase, see section 6.1)

– **Confidence**: (name of the sample): confidence score between 0 (low confidence) and 1 (high confidence) that indicates the confidence in the taxon being present in the sample. The reported confidence score is \(1 - p\)-value under the null hypothesis that the reads map at random positions in the database. The process of randomly mapping reads can be described as a Bernoulli process, and the p-value can then be evaluated for each reference using the number of reads mapping to that reference. In most cases the confidence score will be 1.

– **Coverage**: (name of the sample): coverage estimate for the 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 of a taxon in a specific sample and the amount of coverage in the sample). You can also combine absolute counts and relative abundances by taxonomic levels by selecting the appropriate taxonomic level in the **Aggregate feature** drop-down menu. Incomplete taxonomies at a given level of Aggregation can be hidden using the “Hide incomplete taxonomy” check box.

Finally, if you have previously annotated your table with Metadata (see section 8.7), 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. 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 table, than 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 6.5) and Stacked Area Charts (figure 6.6), 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 (保存/还原设置)), the settings can be saved and applied in other plots, allowing visual comparisons across analyses.

- **Zoomable Sunbursts** (自定义聚焦太阳射线图) 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 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 6.7).

![Sunburst view](image)

Figure 6.7: 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.

### 6.1.2 Taxonomic Profiling report

The Taxonomic Profiling report is divided in three sections (figure 6.8), with an additional fourth section if the option "Auto-detect Paired Distances" was selected.

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

- **Classification of reads** summarizes the number of reads in the sample and the number of uniquely mapping reads that match either the reference database, the host genome or were unclassified.

- **Reference database Summary** gives an overview of the reference database used with the Taxonomic Profiler. A subsection is added in case any duplicates are found in the database. In case of duplicates, only one of the duplicated references is used.

- **Auto-detect Paired Distances** gives the paired distance range.
6.1.3 Sorted Reads Files

The taxonomic profiling tool can sort reads based on their assigned position in the databases (figure 6.9): database matches, host matches, and unclassified reads.

The resulting files are sequence lists that can be visualized as table. The host and database matches tables include a value called **Mapping score**, and calculated for each read at their assigned positions (see [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Mapping_parameters).
In addition, the database matches table provide the **Assigned taxonomy** for the read, taxonomy that was assigned to the read by mapping it to the database.

### 6.2 Workflows

The **Workflows** folder contains three ready-to-use workflows that can quickly get you started with your whole metagenome taxonomic analysis. You can use the workflows as they are, or use them as templates for creating custom or more advanced workflows. In order to copy and edit the workflows, right click the workflow you want and choose **Open Copy of Workflow**.

#### 6.2.1 Data QC and Clean Host DNA

The **Data QC and Clean Host DNA** workflow performs trimming of reads, creates a QC report and cleans the dataset from host DNA, leaving back only the reads that do not match the host genome.

To run the Data QC and Clean Host DNA workflow, go to **Microbial Genomics Module** | **Metagenomics** | **Taxonomic Analysis** | **Workflows** | **Data QC and Clean Host DNA**.

You can select one or several read files to analyze (figure 6.10). When choosing several read files, they will be considered as belonging to one single sample unless the batch mode option is checked, in which case each file will be considered as an individual sample.

![Figure 6.10: Select the reads.](image)

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 6.11).

![Figure 6.11: You can choose to trim adapter sequences from your sequencing reads.](image)

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, select the reference database you will use to map the reads (figure 6.12). It is possible to "Filter host reads". You must then specify the host genome (in the case of human microbiota, the Homo sapiens hg38 for example).

![Select the reference database, and potentially choose to filter against an host genome to remove possible contamination.](image)

The workflow will output a sequence list with reads cleaned from host DNA. 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 [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Qc_Sequencing_Report_Content.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Qc_Sequencing_Report_Content.html). For the trimming report, see [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html).

### 6.2.2 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:

**Metagenomics** | **Taxonomic Analysis** | **Workflows** | **Data QC and Taxonomic Profiling**.

You can select one or several read files to analyze (figure 6.13). When choosing several read files, they will be considered as belonging to one single sample unless the batch mode option is checked, in which case each file will be considered as an individual sample.

![Select the reads to analyze.](image)
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 6.14).

Figure 6.14: 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 6.15), 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 hg38 for example).

Figure 6.15: 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, a coverage estimate, a confidence score for presence of that node. The table can be visualized using the Stacked bar charts and stacked area charts function, as well as the Sunburst charts (see section 7.5.1).

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 taxons 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** gives an overview of the reference database used with the Taxonomic Profiler. 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 [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Qc_Sequencing_Report_Content.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Qc_Sequencing_Report_Content.html). For the trimming report, see [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Trim_output.html).

### 6.2.3 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: **Metagenomics | Taxonomic Analysis | Workflows | Merge and Estimate Alpha and Beta diversities**.

In the first step, select several abundance tables (figure 6.16).

![Figure 6.16: Select abundance tables.](image)

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 8.2 and section 8.3.

The Merge and Estimate Alpha and Beta Diversities workflow generates the results seen in figure 6.17.

![Figure 6.17: Results from the Merge and Estimate Alpha and Beta Diversities workflow.](image)

Please refer to section 8.2 and section 8.3 to learn more about interpreting these results.
Chapter 7

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 (see for example "An introduction to the Pfam protein families database", [http://pid.nci.nih.gov/2011/110913/full/pid.2011.3.shtml](http://pid.nci.nih.gov/2011/110913/full/pid.2011.3.shtml) for further information).

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 4). The resulting contigs can then be annotated with coding sequences (CDS) using the third-party MetaGeneMark plugin. 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 Databases 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 (7.5).

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 8.1). The functional profile of multiple samples can now be visualized and compared as described in Section 5.2.3.

Please Note that the functionality of the tools included in the Functional analysis folder is in beta. As this is still a very active research area, the software is accordingly also under active development and subject to change without notice.
7.1 Find Prokaryotic Genes (beta)

The Find Prokaryotic Genes (beta) tool allows you to annotate a DNA sequence with CDS information. The tool is currently in beta as it is now only tailored for use with near-complete single genomes, and not for metagenome 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 http://ccb.jhu.edu/papers/glimmer3.pdf) and currently consolidate in one tool both build-icm and glimmer3. The standard version of Glimmer is available at https://ccb.jhu.edu/software/glimmer/.

To start the analysis, go to:

Metagenomics | Functional Analysis | Find Prokaryotic Genes (beta)

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 and the "Assembly_ID" annotations are present, the tool will build a separate model for each assembly. The tool can also be run in batch mode.

In the second dialog, it is possible to set the following parameters:

- **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.

- **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.

The tool will output a copy of the input sequence with CDS and Gene annotations.

7.2 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:

Metagenomics | 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.
• **Expectation value.** The minimum expectation value (E) threshold to use.

**Note** that choosing a very large BLAST database with millions of sequences (e.g. the nt, nr and refseq_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 7.1):

- **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.

![Figure 7.1: BLAST Best Hit annotations added to gene cds4 of h. pylori.](image-url)

The tool can also output an annotation table summarizing information about the annotations added to the sequence list.
7.3 Annotate CDS with Best DIAMOND Hit

The **Annotate CDS with Best DIAMOND Hit** tool will allow 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. 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.

The version of the DIAMOND binaries bundled with the tool is v0.9.21. For questions or comments about DIAMOND, see [https://github.com/bbuchfink/diamond](https://github.com/bbuchfink/diamond).

To start the tool, go to:

**Metagenomics** | **Functional Analysis** | **Annotate CDS with Best DIAMOND Hit**

Several parameters are available:

- **Genetic code parameters.** The genetic code used for translating CDS to proteins.
- **DIAMOND parameters.**
  - **Expectation value.** The minimum expectation value (E) threshold to use.
  - **Protein Sequence Database.** Popular public protein sequence databases can be downloaded using the Download Protein Database tool (see section 17.3).

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.

7.4 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:

**Metagenomics** | **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 17.4). 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 [http://geneontology.org/page/go-slim-and-subset-guide](http://geneontology.org/page/go-slim-and-subset-guide).

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 7.2. The annotation contains the following fields:

- **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.

![Figure 7.2: Pfam and GO annotations added to gene cds4 of h. pylori.](image-url)
• **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.

### 7.5 Build Functional Profile

To compute the number of reads in a sample mapping to regions involved with Pfam domains or GO terms, you can run the Build Functional Profile tool by going to:

Metagenomics | Functional Analysis | Build Functional Profile

The following parameters can then be set:

• **Reference.** A reference set of contigs annotated with Pfam domains and/or BLAST hits. If the read mapping contains an annotated genome, this parameter is optional.

• **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 17.4).

• **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 [http://geneontology.org/page/go-slim-and-subset-guide](http://geneontology.org/page/go-slim-and-subset-guide).

• **Propagate GO mapping.** When selected, Pfam annotations are mapped to the relative GO terms and all their more general 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".

You can then select which output elements should be generated.

• **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, as preexisting GO annotations on pfam domains are ignored by the tool.

• **Create BLAST hit functional profile.** Abundance table obtained by counting reads overlapping BLAST 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 or best BLAST hit.

### 7.5.1 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 7.3)

![Figure 7.3: Functional profile abundance table.](image)

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 function 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 8.7), 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. 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 table, than 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 7.4) and Stacked Area Charts (figure 7.5), 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.

![Stacked bar chart](image)

*Figure 7.4: Stacked bar chart.*

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 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 7.6).

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 7.6: Sunburst view.
Chapter 8

Abundance analysis

8.1 Merge Abundance Tables

The Merge Abundance Table tool allows you to merge together abundance tables for different samples. The input can be OTU, functional, or whole metagenome taxonomic profiling abundance tables but not a mix of these. The output will be a merged table of the same nature as the ones input in the tool. The merged abundance table will include the metadata that was originally in the input abundance tables.

If you select Create Report, the tool will create a summary report table. For each input abundance table, the report will contain the name of the abundance table, the number of samples in the table and the number of features contained in the table.

Merged abundance tables can be used as input for various tools:

- Alpha Diversity, see section 8.2
- Beta Diversity, see section 8.3
- Differential Abundance Analysis, see section 8.5
- Create Heat Map for Abundance Table, see section 8.6

8.2 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 the following three tools: 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 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

Microbial Genomics Module | Metagenomics | Abundance Analysis

Choose an abundance table to use as input. The next wizard window offers you to set up different analysis parameters (figure 8.1). For example, you can select which diversity measures to calculate (see section 8.2.1), and parameterize the rarefaction analysis. If you are working with OTU abundance tables, 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 Alpha Diversity tool.

The rarefaction analyses are done differently depending on the type of abundance table used as input. For OTU and functional abundance tables, where abundances are counts, rarefaction is calculated by sub-sampling the abundances in the different samples at different depths. For whole metagenome taxonomic profiling abundance tables, where abundances are coverage estimates, sub-sampling is not possible, so diversity is estimated using a probabilistic model corresponding to our qualification criteria instead (see section 6.1).

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.

- **Replicates at each depth** (for counts-based abundance tables only). How many times the algorithm sub-samples the data at each depth.
- **Sample with replacement** Whether the sampling should be performed with or without replacement.

The tool will generate a graph for each selected Alpha diversity measure (figure 8.2). 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.

![Alpha Diversity graph based on phylogenetic diversity](image)

Figure 8.2: An example of Alpha Diversity graph based on phylogenetic diversity.

Note that the option "Show derived legend info" is enabled by default (figure 8.3). 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.

### 8.2.1 Alpha diversity measures

The available diversity measures are:

- **Total number**: The number of features (e.g. OTUs when doing out clustering, GO terms when building functional profiles or organisms when performing taxonomic profiling) observed in the sample.
- **Chao 1 bias-corrected**: \( \text{Chao1-bc} = D + \frac{f_1(f_1-1)}{2(f_2+1)} \).
Figure 8.3: 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.

- Chao 1: \( \text{Chao1} = D + \frac{f_1^2}{2f_2} \).

- Simpson’s index: \( \text{SI} = 1 - \sum_{i=1}^{n} p_i^2 \).

- Shannon entropy: \( \text{H} = \sum_{i=1}^{n} p_i \log_2 p_i \).

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.

8.3 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 8.3.1). 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 (\( \text{PCoA} \)) in the bottom of the Beta Diversity results (\( \text{PCoA} \)).
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

**Microbial Genomics Module** | **Metagenomics** | **Abundance Analysis** | **Beta Diversity**

Select an abundance table with more than one sample as input (i.e., an OTU table table, or a merged functional or profiling table) and set the parameters for the beta diversity analysis as shown in figure 8.4.

![Figure 8.4: Set up parameters for the Beta diversity tool.](image)

The output of the tool is a 3D PCoA plot (figure 8.5) that can also be seen as a table.

![Figure 8.5: Beta diversity results seen as a 3D PCoA.](image)

Use the settings in the right hand side panel to explore the results and visualize them adequately.
8.3.1 Beta diversity measures

The following beta diversity measures are available:

- **Bray-Curtis**: \( B = \frac{\sum_{i=1}^{n} |x^A_i - x^B_i|}{\sum_{i=1}^{n} (x^A_i + x^B_i)} \)

- **Jaccard**: \( J = 1 - \frac{\sum_{i=1}^{n} \min(x^A_i, x^B_i)}{\sum_{i=1}^{n} \max(x^A_i, x^B_i)} \)

- **Euclidean**: \( E = \sum_{i=1}^{n} \sqrt{(x^A_i - x^B_i)^2} \)

where \( n \) is the number of OTUs and \( x^A_i \) and \( x^B_i \) are the abundances of OTU \( i \) in samples \( A \) and \( B \), respectively.

The following distances are also available:

- **Unweighted UniFrac**: \( d(U) = \frac{\sum_{i=1}^{n} b_i |I(p^A_i > 0) - I(p^B_i > 0)|}{\sum_{i=1}^{n} b_i} \)

- **Weighted UniFrac**: \( d(W) = \frac{\sum_{i=1}^{n} b_i |p^A_i - p^B_i|}{\sum_{i=1}^{n} b_i (p^A_i + p^B_i)} \)

- **Weighted UniFrac not normalized**: \( d(w) = \sum_{i=1}^{n} b_i |p^A_i - p^B_i| \)

- **D_0 UniFrac**: The generalized UniFrac distance \( d^{(0)} = \frac{\sum_{i=1}^{n} b_i |p^A_i - p^B_i|}{\sum_{i=1}^{n} b_i} \)

- **D_0.5 UniFrac**: The generalized UniFrac distance \( d^{(0.5)} = \frac{\sum_{i=1}^{n} b_i \sqrt{p^A_i + p^B_i} |p^A_i - p^B_i|}{\sum_{i=1}^{n} b_i \sqrt{p^A_i + p^B_i}} \)

where \( n \) is the number of branches in the phylogenetic tree, \( b_i \) is the length of branch \( i \); \( p^A_i \) and \( p^B_i \) are the proportion of taxa descending from branch \( i \) for samples \( A \) and \( B \), respectively; and the indicator functions \( I(p^A_i > 0) \) and \( I(p^B_i > 0) \) 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].

8.4 PERMANOVA Analysis

PERMANOVA Analysis (PERmutational Multivariate ANalysis Of VAriance, also known as non-parametric 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:

Microbial Genomics Module | Metagenomics | Abundance Analysis | PERMANOVA Analysis

Choose an abundance table with more than one sample as input (i.e., an OTU table table, or a merged functional or profiling 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 8.6), you can choose which Beta diversity measure to use (see section 8.3.1 for definitions). 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).

![Figure 8.6: Beta diversity and Phylogenetic diversity measures are included in the PERMANOVA analysis.](image)

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.
8.5 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 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: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=_statistical_model.html.

To run the tool:

Microbial Genomics Module ( )  |  Metagenomics ( )  |  Abundance Analysis ( )  |  Differential Abundance Analysis ( )

Select an abundance table with more than one sample as input (i.e., an OTU table, or a merged functional or profiling table), and if you want to test differential abundance based on metadata defined groups of samples (figure 8.7). 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 or between all group pairs.

![Figure 8.7: Specify an abundance table and other parameters.](image)

The tool generates a Venn diagram for three pairwise comparisons at a time (figure 8.8). 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.

The values included in the table for each pairwise comparison are:

- **Max group means** For each group in the statistical comparison, the average RPKM is calculated. This value is the maximum of the average RPKM’s.
- **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.

### 8.6 Create Heat Map for Abundance Table

The hierarchical clustering groups features by the similarity of their genomes over the set of samples, and clusters samples by the similarity of genomes over their features. Each clustering has a tree structure that is generated as follow:

1. The tool considers each feature or sample to be a cluster.
2. It calculates pairwise distances between all clusters, and join the two closest clusters into one new cluster.
3. This process is repeated until there is only one cluster left, which contains all the features or samples.
4. The tree is then drawn so that the distances between clusters are reflected by the lengths of the branches in the tree.
The Create Heat Map for Abundance Table tool uses the TMM normalization described in (described in section ??) to make samples comparable, then does a z-score normalization to make features comparable.

To create a heat map:

Select an abundance table with more than one sample as input (i.e., an OTU table, or a merged functional or profiling table) and specify a distance measure and a cluster linkage (figure 8.9). The distance measure is used to specify how distances between two features or samples should be calculated. The cluster linkage specifies how the distance between two clusters, each consisting of a number of features or samples, should be calculated. Learn more about how distances and clusters are calculated at http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Clustering_features_samples.html.

![Figure 8.9: Select an abundance table.](image-url)

After having selected the distance measure, set up the feature filtering options (figure 8.10).

Indeed, genomes usually contain too many features to allow for a meaningful visualization. Clustering hundreds of thousands of features is also very time consuming. Therefore it is recommend to reduce the number of features before clustering and visualization. There are several different filter settings:

- **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
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.

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 8.11).

In order to create a heat map with a specific taxonomic level information, it is possible to use the
option "Aggregate feature" in the right hand side panel of the Abundance table. When aggregating
an abundance table, by class for example , a new column called "Class (Aggregated)" containing
the class names is now created. This name will then be used when creating a Heat Map. This
is done in order to avoid very long feature names in abundance tables and downstream analysis
tools.

8.7 Add Metadata to Abundance Table

It is useful for abundance tables to be decorated with metadata on the samples. This can be
done by importing metadata and associating it with the reads before generating an abundance
table. But it is also possible to do it after the abundance table was generated using the Add
Metadata to Abundance Table tool.

To run the tool go to: 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 8.12).

Figure 8.12: 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.

8.8 Convert Abundance Table to Experiment

Most statistical tools take abundance tables as input, but the ones listed below require an experiment table:

- **Empirical Analysis of DGE ( )**: Please use now the Differential Abundance Analysis.

- **Hierarchical Clustering of Samples ( )** with which the data needed to be normalized beforehand. Please use now the Create Heat Map for Abundance Table that does not require any normalization prior to its use.

The Convert Abundance Table to Experiment allows you to transform an abundance table into an experiment table, allowing users to use the thus-generated table to perform the statistical tests listed above.

To use the tool, go to:

Microbial Genomics Module | Metagenomics | Abundance Analysis | Convert Abundance Table to Experiment

Choose an abundance table as input, and define which metadata group is to be considered as factor.

The tool output is a table labeled (experiment). The first column is the name of the group used as factor in the analysis (its taxonomy and an ID number). For each feature (e.g. an OTU, an organism, a Pfam family or a GO term) there will be the following data:

- **Range**: The difference between the highest and the lowest expression value for the feature over all the samples.

- **IQR**: The inter-quantile range of the values for a feature across the samples, that is, the difference between the 75%-ile value and the 25%-ile value.

- **Difference**: The difference for a two-group experiment between the mean of the expression values across the samples assigned to group 2 and the mean of the expression values across the samples assigned to group 1. Thus, if the mean expression level in group 2 is higher than that of group 1 the 'Difference' is positive, and if it is lower the 'Difference' is negative. For experiments with more than two groups the 'Difference' contains the difference between the maximum and minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs before the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

- **Fold change**: For a two-group experiment the 'Fold Change' tells you how many times bigger the mean expression value in group 2 is relative to that of group 1. If the mean expression
value in group 2 is bigger than that in group 1 this value is the mean expression value in group 2 divided by that in group 1. If the mean expression value in group 2 is smaller than that in group 1 the fold change is the mean expression value in group 1 divided by that in group 2 with a negative sign. Thus, if the mean expression levels in group 1 and group 2 are 10 and 50 respectively, the fold change is 5, and if the mean expression levels in group 1 and group 2 are 50 and 10 respectively, the fold change is -5. For experiments with more than two groups, the ‘Fold Change’ column contains the ratio of the maximum of the mean expression values of the groups to the minimum of the mean expression values of the groups, multiplied by -1 if the group with the maximum mean expression value occurs before the group with the minimum mean expression value (with the ordering: group 1, group 2, ...).

- **Taxonomy**: The taxonomy of the feature, if specified. For OTUs, this is specified by the reference database when a database entry was used as Reference.

- **Expression values and Means for each site**: The expression values represents the number of reads belonging to the feature in a specific sample, and Means is the mean of these expression values.

You can create a sub-experiment by selecting only some of the features from your experiment table (the most abundant ones for example).
Part III

Typing and Epidemiology (beta)
Chapter 9

Introduction to Typing and Epidemiology (beta)

Note that the functionality of the plugin described within this section is in beta. It is under active development and subject to change without notice.

Next generation sequencing (NGS) data from whole pathogen genomes are 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 ready-to-use workflows for routine surveillance or outbreak analysis of a specific pathogen (figure 9.1).

![Figure 9.1: Typing of cultured microbial samples shown as process diagram.](image)

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 features provided within the Typing and Epidemiology (beta) tools and workflows are described in the following chapters:

- Description of the workflows templates and how to customize them to fit your analyses in chapter 11.
- How to import or create a Metadata Table, associate it with elements such as sequencing reads and use a Metadata Result Table to access to your results and start additional analyses in chapter 10.
- Description of the individual tools from the module.
- Creation and visualization of phylogenetic SNP or K-mer trees in chapter 14.
Chapter 10

Handling of metadata and analysis results

When typing and/or performing epidemiological studies of microbes, sample information such as taxonomy, date of isolation, original source of the sample, etc, may be relevant for data interpretation. Once a CLC Genomics Workbench version 11.0 or higher, or the Biomedical Genomics Workbench 5.0 or higher is installed on your computer, CLC Microbial Genomics Module facilitates the organization and management of such additional data elements (metadata) in a tabular form. In addition, CLC Microbial Genomics Module facilitates the organization and management of generated analysis results and enables these to be shown as decorations on phylogenetic trees.

10.1 Importing Metadata Tables

The basic Metadata import tool is fast and easy. General features of this importer are:

- The importer takes Excel (*.xlsx/*.xls) format files. All columns will be imported as text columns.
- The first column in the Excel file must have unique entries (it is designated as the key column) and these entries must match (exactly or partially) data element names to allow association between the elements and the metadata.
- Data elements can have metadata associated as part of the import or later.
- If metadata association is successful, visual feedback about the elements associated with metadata is provided in the wizard.

To run the basic importer, go to:

File | Import (曙) | Import Metadata (曙)

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

The rows in the spreadsheet are displayed in the Metadata preview table, as shown in figure 10.1.
In the second wizard window, you can choose to associate data elements to the metadata. Select in the field **Location of data** the data elements to associate. In the Matching scheme area, select whether data element names must match exactly or partially the entries in the first column of the metadata (figure 10.2).

You can now click on the button Next to select where you wish the metadata table to be saved.

The associated information can be viewed for a given data element in the Show Element Info view, as show in figure 10.3.

If desired, a metadata table can be edited later from within the Metadata Table editor (see section 10.2). It is also possible to create a metadata table directly within the workbench. For more information on Advanced Metadata Import, see http://resources.
10.2 Associating data elements with metadata

Once a Metadata Table has been saved, data elements - such as, but not restricted to, sequencing reads - may be associated with its rows if it was not already done during import. Association means that you can keep track of the sample to which a particular analysis result pertains without having to resort to data element naming conventions or an elaborate folder structure.

Each association between a particular data element and a row in your Metadata Table will be qualified by a "role" label that indicates what the role of the data element is with respect to your row. For example, a suitable role for a freshly imported data element would be "Sample data". Each analysis tool will subsequently provide its own role labels when transferring the results to the table. As an example, a read mapping tool will assign the role "Un-mapped reads" to the Sequence List it produces, allowing you to keep track of which Sequence List contained the original imported NGS reads, and which contained the un-mapped ones.

To associate read- and metadata automatically: Click **Associate Data** and select **Associate Data Automatically**. When using the Associate Data Automatically option, you will be able to select the read files you would like to associate with the metadata (figure 10.4).

![Figure 10.4: When using the Associate Data Automatically option, the wizard will search among the selected elements (47 in this example) and select the ones with names identical with the Key identifier (here 14).](image)

The role assignment for this association is set by the wizard to "Sample data" (figure 10.5) but can be edited to fit your needs.

![Figure 10.5: Specify the role label for freshly imported data elements.](image)
Sometimes, more data elements correspond to a specific metadata key, or the names of the elements do not match the metadata key value. In these cases, you must associate a given metadata role to one or more data elements manually: Click **Associate Data** and select **Associate Data with row** ( ), or right click on the row you would like to process. The wizard is the same as the one described above, the difference being that you can only create one association at a time.

When you are done with associating sequencing data and metadata, you can open the sequencing file and check the icon **Show Element Info** ( ) to get a process log that indicates the association between Metadata Table rows and data was successful (figure 10.6).

![Figure 10.6: The Show Element Info button displays association between the element and the metadata.](image)

**Tips when using Metadata Tables:**

- A **new metadata table must be saved before the association** of data elements with metadata is possible.

- Key Column (the unique identifier) information and the name of the sequence data has to be **perfectly identical** for the automatic association to function (a partial matching scheme refer to the length of the identifier).

- **Do not move/import/export** the Metadata Table to a new file location, as the Result Metadata Table then loses its references to the Metadata Table, and the data will still be associated to the old and nonexistent file.

- **Do not associate the same read file to multiple metadata tables**, as this will inhibit the ability to create SNP tree(s) with this file.

### 10.3 Create a Result Metadata Table

Once the Metadata Table rows are associated to sample data, it is possible to generate a **Result Metadata Table**. Go to:
A dialog as shown in figure 10.7 is then displayed. Select the Metadata Table, click Next, select Save, specify location and click Finish.

![Figure 10.7: Creation of a Result Metadata Table from a Metadata Table.](image)

A new Result Metadata Table has now been created.

When first opening the created Result Metadata Table, it is empty, as no analysis results have yet been generated (figure 10.8).

![Figure 10.8: The created Result Metadata Table is empty as no analysis results has been produced yet.](image)

Novel (= not yet analyzed) samples can be added manually to the table by clicking the Add Novel Samples button. The following message will appear: Any available novel samples have now been added and all available novel sample(s) will be listed in the Metadata Result Table. In yellow is listed the available Metadata associated to the Novel samples, while analysis Result Metadata fields are still empty. Figure 10.9 is an example were a single novel sample has been added to the analysis Result Metadata Table.

If for some reason Result Metadata row(s) 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 10.10.

In most cases, analysis results will be added automatically to the Result Metadata Table when using a properly designed workflow or certain tools from the module. It is also possible to add manually generated analysis results to the table using the Add to Result Metadata.
CHAPTER 10. HANDLING OF METADATA AND ANALYSIS RESULTS 80

Figure 10.9: By clicking the "Add Novel Samples button", a row including the Metadata of the novel Salmonella sample is now added to the otherwise empty Result Metadata Table.

Figure 10.10: 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.

10.4 Running an analysis directly from a Result Metadata Table

Analysis results from tools listed in the table of section 10.3 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 (http://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 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 10.11, 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).

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 10.4.1). 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.

### 10.4.1 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.

10.4.2 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 (described in section 14.1) is displayed. The read mappings are preselected as input. The variant tracks and the Result Metadata Table are automatically preselected as parameters.

10.5 Add to Result Metadata Table

To add manually results to an existing Result Metadata Table, go to:

Microbial Genomics Module | Typing and Epidemiology (beta) | Result Metadata | Add to Result Metadata

1. In the first wizard window, select the relevant Result Metadata Table (see figure 10.12) and click Next.
CHAPTER 10. HANDLING OF METADATA AND ANALYSIS RESULTS

2. Now select the relevant Result object(s) to be added to the Result Metadata Table (see figure 10.13) and click **Next**.

3. Finally, select to **Save** and click **Finish**.

The output of this tool are generated as new columns in your Metadata Result Table (figure 10.14).

Figure 10.12: *First select the Result Metadata Table you want to add results to.*

Figure 10.13: *In second step of this example, the identified best matching sequence for a particular sample is added.*

Figure 10.14: *Extra column are added to the Result Metadata Table, including the data for the particular sample that was specified in step 1.*
10.6 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 10.15).

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 10.16).

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 10.17). 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 10.17.

Figure 10.16: Specification of the read mapping to be associated with genome metadata.

Figure 10.17: Filter for elements who share a tick in the column newly generated by the Use Genome as Result tool
Chapter 11

Workflow templates

CLC Microbial Genomics Module offers three workflows\(^1\) templates which can be used "as is" or customized for common use cases. The following workflows can be run for single samples or in batch. In the first case, selecting several items to be analyzed will run them as a being one sample and will produce **one set of output files for all items**. Running in batch allows you to select several items and create **as many set of output files as items you selected**.

To explore the content and parameters of a template workflow, select the workflow, right click on its name and select the choose the option **Open Copy of Workflow**. You can pre-set and save parameters and input files directly from the copy of the workflow in the View Area, which is useful is you intend to run the exact same workflow repeatedly. Otherwise just double-click on the workflow’s name in the Navigation Area and follow the instructions on each wizard step.

The following sections describe how to configure the provided three workflow templates that are recommended as the starting points for your typing- and epidemiological analyses. Relevant example data can be downloaded and imported by following the tutorial on **Typing and Epidemiological Clustering of Common Pathogens** (see [http://resources.qiagenbioinformatics.com/tutorials/Typing_Epidemiological_Clustering.pdf](http://resources.qiagenbioinformatics.com/tutorials/Typing_Epidemiological_Clustering.pdf)).

**Note!** The typing and epidemiology template workflows automatically associate the analysis results to a Result Metadata Table specified by the user. Searching and quick filtering of the sample metadata and the generated analysis result data enable the selection of data subsets that can be in turn used for additional downstream SNP tree analyses.

### 11.1 Map to Specified Reference

Once analysis has been performed using the **Type a Known Species** or the **Type among Multiple Species** workflow, the best matching reference is listed in the Result Metadata table (figure 11.1, see column Best match).

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

Figure 11.1: *Best match references are listed for each row in the Result Metadata Table.*

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 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 12.2.

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 14.2.

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 11.2) that will perform a re-mapping of the reads followed by variant calling.
Figure 11.2: Overview of the template Map to Specified Reference workflow.
11.1.1 How to run the Map to Specified Reference workflow on a single sample:

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

Microbial Genomics Module ( ) | Typing and Epidemiology (beta) ( ) | Workflows ( ) | Map to Specified Reference ( )

1. Specify the sample you would like to type (figure 11.3) and click **Next**.

![Figure 11.3: Select the reads from the sample you would like to type.](image)

2. Specify the **Result Metadata Table** you want to use (figure 11.4) and click **Next**.

![Figure 11.4: Select the metadata table you would like to use.](image)

3. 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 11.5).

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.

---

4. Specify the parameters for the Maps Reads to Reference tool (figure 11.6) using 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.

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). \textbf{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**.

5. Specify the parameters for the **Basic Variant Detection** tool (figure 11.7) before clicking **Next**.

![Figure 11.7: Specify the parameters to be used for the Basic Variant Detection tool.](image)

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.

6. 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**.

Three outputs are generated (figure 11.8):

![Output files from the Map to Specified Reference workflow](Image)

**Figure 11.8**: Output files from the Map to Specified Reference workflow.

• **Reads Track**: output from the Local Realignment tool

• **Variants 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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

You have now the data necessary to create a SNP tree for your samples as explained in chapter 14.1.

### 11.1.2 How to run the Map to Specified Reference workflow on a batch of samples:

To be able to run multiple sample data sets in **batch mode**, the user must initially make a copy of the template workflow, specify a Result Metadata Table and save the copy of the workflow in the Navigation Area before running it.

1. Select the workflow Map to Specified Reference in the toolbox, right-click on the name and choose the option **Open Copy of Workflow** (figure 11.9).

![Figure 11.9: Open a copy of a workflow.](image)

2. This opens a copy of the workflow in the view area of your workbench. Double click on the green tile representing the Result Metadata Table input file (highlighted in red in figure 11.10).

![Figure 11.10: Double click on the green Result Metadata Table input file tile (highlighted in red).](image)

3. It opens a window where you specify the **Result Metadata Table** created as specified in section 11.3.1 (figure 11.11). Click on **Finish**.
4. Save the workflow in your Navigation Area by pressing ctrl+S or by right-clicking on the tab and selecting **File | Save as** followed by the file location of your choice. Alternatively, you can also simply drag the tab to the relevant location in your Navigation Area.

5. You can now click on the button **Run** at the bottom of the copy of the workflow in the View Area (highlighted in red in figure 11.12).

6. Check the option **Batch** (highlighted in red in figure 11.13) before selecting several items (samples or folder(s) of samples) to be analyzed. Click **Next**.

7. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you actually want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 11.14).

8. In the third wizard window, you can see that the Result Metadata Table you specified earlier
Figure 11.14: Choose which of the samples present in the selected folder(s) you want to analyze.

is already selected. Check that it is indeed the Result Metadata Table you intended to use and click Next.

9. The rest of the workflow is similar to the one described in section 11.1.1. Refer to this section to understand what parameters can be set, and which outputs are generated.

10. In the last Result Handling window, we recommend saving the results in separate folders.

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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

11.2 Type Among Multiple Species

The Type among Multiple Species workflow is designed for typing a sample among multiple predefined species (figure 11.15).

Similar to the Type a Known Species workflow, it allows identification of the closest matching reference species among the user specified reference list(s), but in this case, the list(s) 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 10.4.1.

11.2.1 Preliminary steps to run the Type Multiple Species workflow

Before starting the workflow,

- Download microbial genomes using either the Download or Set Up Microbial Reference Database tools or the Download Pathogen Reference Database tool (see section IV).
- Download the MLST schemes using the Download MLST Schemes tool (see section 19.1).
Download the Database for the Find Resistance tool using the Download of Database for Find Resistance tool (see section 17.1).

Create a New Result Metadata table using the Create Result Metadata Table tool (see section 10.3).

Your navigation area should now look similar to the figure 11.16.

11.2.2 How to run the Type among Multiple Species workflow for a single sample

To run the workflow on a single sample containing multiple species, go to

Microbial Genomics Module | Typing and Epidemiology (beta) | Workflows | Type among Multiple Species

1. Specify the sample you would like to type (figure 11.17) and click Next. Remember that even if you select several items, they will be considered as being part of the same sample.

2. Specify the Result Metadata Table you want to use (figure 11.18) and click Next.
3. You can specify a trim adapter list\(^3\) 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 11.19).

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.
- **Also search on reversed sequence**: the adapter sequences will also be searched on reverse sequences.

Click **Next**.

4. Choose the **species-specific references** to be used by the **Find Best Matches using K-mer Spectra** tool (figure 11.20). The list can be a fully customized list, the downloaded bacterial genomes from NCBI list (see section 16.3.1) or a subset hereoff. Click **Next**.

5. 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 11.21).

6. Specify the **resistance database** (figure 11.22) and set the parameters for the **Find Resistance** tool.

The parameters that can be set are:

- **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.
Figure 11.21: Specify the schemes that describe best your sample.

Figure 11.22: Specify the resistance database to be used for the Find Resistance tool.

- **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 that agree on contig, source and predicted phenotype where one is contained by the other.

Click **Next**.

7. Specify the parameters for the **Basic Variant Detection** tool (figure 11.23) before clicking **Next**.

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.
Figure 11.23: Specify the parameters to be used for the Basic 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.

8. Specify the value for which you would like the **Identify MLST** tool to report low coverage (figure 11.24). Click **Next**.

![Figure 11.24](image)

**Figure 11.24**: Specify when you would like low coverage to be reported for the Identify MLST tool.

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 to this effect) and click on the button labeled **Finish**.

The following outputs are generated. You can find them all in the new folder you created to save them (figure 11.25), but those marked with a (*) have also been added automatically to the New Metadata Result Table (see section 11.2.3 to understand where your results have been saved).

![Figure 11.25](image)

**Figure 11.25**: Output files from the Type Among Multiple Species workflow.


• **Trimmed sequences**: list of the sequences that were successfully trimmed.

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

- **Contig list resistance table**: result table from the *Find Resistance* tool, reports the found resistance.

- **Quality 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.

- **Consensus NGS MLST Report**: output from the *Identify MLST* tool, includes information on which MLST scheme was applied, the identified sequence type (ST) as well as an overview table which summarizes the targeted genes, their consensus sequence profile and coverage.

- **Read Tracks, Genome Track and/or Annotation Track**: 2 outputs with the appendix "cut" or "typed regions cut" from the *Extract Regions from Tracks* tool.

- **Track List**: output from the *Create Track List* tool.

- **Variants 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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

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.

### 11.2.3 Example of results obtained using the Type among Multiple Species workflow

The following example includes typing of 5 samples: 2 *Salmonella enterica* (acc nos ERR274480, ERR277203), 1 *Listeria monocytogenes* (acc no DRR000982) and 2 *Shigella* (acc nos ERR279274, ERR279296). Using the above customization of the *Type among Multiple Species* workflow, analysis results are automatically summarized in the Result Metadata Table as shown in figure 11.26. The analysis results in this example include the name and the description of the best matching reference, the identified MLST and applied scheme, and some cases where aminoglycoside resistance was found. This is not the case in this example, but you could also
choose (using options in the Table Setting window) to display additional information such as the ‘applied best match’ and ‘resistance databases’ for example.

By selecting samples in the Result Metadata Table, additional analyses can be performed directly from this Table:

- Generation of SNP trees based on the same reference used for read mapping and variant detection (section 14.1).
- Run validated workflows (workflows that are associated with a Result Metadata Table and saved in your Navigation Area).

### 11.2.4 How to run the Type among Multiple Species workflow on a batch of samples:

To be able to run multiple sample data sets in batch mode, the user must initially make a copy of the template workflow, specify a Result Metadata Table and save the copy of the workflow in the Navigation Area before running it.

1. Select the workflow Type among Multiple Species in the toolbox, right-click on the name and choose the option **Open Copy of Workflow** (figure 11.27).

2. This opens a copy of the workflow in the view area of your workbench. Double click on the green tile representing the Result Metadata Table input file (highlighted in red in figure 11.28).

3. It opens a window where you have to specify the **Result Metadata Table** you created for this particular workflow (figure 11.29). Click on **Finish**.
Figure 11.27: Open a copy of a workflow.

Figure 11.28: Double click on the Result Metadata Table green input file tile.

Figure 11.29: Specify the Result Metadata Table you created for running this workflow (here called New metadata table results).

4. Save your workflow in your Navigation Area.

5. You can now click on the button Run at the bottom of the copy of the workflow in the View Area (highlighted in red in figure 11.30).

Figure 11.30: Open the copy of the workflow from the Navigation Area and start running it by clicking on the button labeled Run at the bottom of the View Area.

6. Check the option Batch (highlighted in red in figure 11.31) before selecting several items (samples or folder(s) of samples) to be analyzed. Click Next.

7. The next wizard window gives an overview of the samples present in the selected folder(s).
CHAPTER 11. WORKFLOW TEMPLATES

Figure 11.31: Remember to tick the button labeled Batch at the bottom of the wizard window before selecting the folders containing the samples you want to analyze.

Choose which of these samples you actually want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 11.32).

Figure 11.32: Choose which of the samples present in the selected folder(s) you want to analyze.

8. In the third wizard window, you can see that the Result Metadata Table you specified earlier is already selected. Check that it is indeed the Result Metadata Table you intended to use and click Next.

9. The rest of the workflow is similar to the one described in section 11.2.2. Refer to this section to understand what parameters can be set, and which outputs are generated.

10. In the last Result Handling window, we recommend saving the results in separate folders.

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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

11.3 Type a Known Species

The Type a Known Species workflow is designed for typing of samples representing a single known species (figure 11.33). It allows identification of the closest matching reference species among the user specified reference sequences, identifies the associated MLST, 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.
11.3.1 Preliminary steps to run the Type a Known Species workflow

Before starting the workflow,

- Download microbial genomes using either the Download or Set Up Microbial Reference Database tools or the Download Pathogen Reference Database tool (see section IV).
- Download the MLST schemes using the Download MLST Schemes tool (see section 19.1).
- Download the Database for the Find Resistance tool using the Download of Database for Find Resistance tool (see section 17.1).
- Create a New Result Metadata table using the Create Result Metadata Table tool (see section 10.3).

When you are ready to start the workflows, your navigation area should look similar to the figure 11.34.

11.3.2 How to run the Type a Known Species workflow for a single sample

To run the workflow on a single sample, go to:
CHAPTER 11. WORKFLOW TEMPLATES

Figure 11.34: Overview of the Navigation area after creating the result metadata table and downloading the databases and MLST schemes necessary to run the workflows.

1. Specify the **sample** you would like to type (figure 11.35) and click **Next**. Remember that even if you select several items, they will be considered as being part of the same sample.

Figure 11.35: Select the reads from the sample you would like to type.

2. Specify the **Result Metadata Table** you want to add your results to (figure 11.36) and click **Next**.

Figure 11.36: Select the metadata table you would like to use.

3. You can specify a **trim adapter list**\(^4\) and set up parameters if you would like to trim your

\(^4\)Learn about trim adapter lists at [http://resources.qiagenbioinformatics.com/manuals/](http://resources.qiagenbioinformatics.com/manuals/)
sequences from adapters. Specifying a trim adapter list is optional but recommended to ensure the highest quality data for your typing analysis (figure 11.37).

![Figure 11.37: You can choose to trim adapter sequences from your sequencing reads.](image)

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.
- **Also search on reversed sequence**: the adapter sequences will also be searched on reverse sequences.

Click **Next**.

4. Choose the **species-specific references** to be used by the **Find Best Matches using K-mer Spectra** tool (figure 11.38). The list can be a fully customized list and/or a subset of the downloaded bacterial genomes from NCBI list (see section 16.3.1). Click **Next**.

![Figure 11.38: Specify the references for the Find Best Matches using K-mer Spectra tool.](image)

5. Specify the **resistance database** (figure 11.39) and set the parameters for the **Find Resistance** tool.

   The parameters that can be set are:
• **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 that agree on contig, source and predicted phenotype where one is contained by the other.

Click **Next**.

6. Specify the parameters for the **Basic Variant Detection** tool (figure 11.40) before clicking **Next**.

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**: 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**.

7. Specify the **MLST scheme** to be used for the Identify MLST Scheme from Genomes tool (section 15.2). This scheme corresponds to the single species included in the reference list(s) (figure 11.41).

![Figure 11.41: Specify the scheme that describes best your sample.](image)

8. 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 11.42), but those marked with a (*) in the list below will also be added automatically to the Metadata Result table.

![Figure 11.42: Output files from the Type a Known Species workflow.](image)

- **Trimmed sequences**: list of the sequences that were successfully trimmed
- **Contig list**: contig list from the De novo assembly tool
- **(*)Contig list resistance table**: result table from the Find Resistance tool, reports the found resistance.
• Quality 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.

• Consensus NGS MLST Report: output from the Identify MLST tool, includes information on which MLST scheme was applied, the identified sequence type (ST) as well as an overview table which summarizes the targeted genes, their consensus sequence profile and coverage.

• Read Tracks, Genome Track and/or Annotation Track: 2 outputs with the appendix "cut" or "typed regions cut" from the Extract Regions from Tracks tool

• Track List: output from the Create Track List tool

• Variants 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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

11.3.3 Example of results obtained using the Type a Known Species workflow

In this section, the Salmonella enterica sample sequence data (acc no ERR274480) was typed using a customized Salmonella enterica version of the Type a Known Species workflow: the applied reference list only included Salmonella specific sequences, the MLST scheme was specified as Salmonella enterica and the created Result Metadata Table included only this particular data set.

Once the workflow analysis was performed on this single dataset, 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 11.43). 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.

11.3.4 How to run the Type a Known Species workflow on a batch of samples:

To be able to run multiple sample data sets in batch mode, the user must initially make a copy of the template workflow, specify a Result Metadata Table and save the copy of the workflow in the Navigation Area before running it.

1. Select the workflow Type a Known Species in the toolbox, right-click on the name and choose the option Open Copy of Workflow (figure 11.44).
2. This opens a copy of the workflow in the view area of your workbench. Double click on the green tile representing the Result Metadata Table input file (highlighted in red in figure 11.45).

3. It opens a window where you have to specify the Result Metadata Table you created for this particular workflow (figure 11.46). Click on Finish.

4. Save your workflow in the Navigation Area.

5. You can now click on the button Run at the bottom of the copy of the workflow in the View Area (highlighted in red in figure 11.47).

6. Check the option Batch (highlighted in red in figure 11.48) before selecting several items (samples or folder(s) of samples) to be analyzed. Click Next.
Figure 11.46: Specify the Result Metadata Table you created for running this workflow (here called New metadata table results).

Figure 11.47: Open the copy of the workflow from the Navigation Area and start running it by clicking on the button labeled Run at the bottom of the View Area.

Figure 11.48: Remember to tick the button labeled Batch at the bottom of the wizard window before selecting the folders containing the samples you want to analyze.

7. The next wizard window gives you an overview of the samples present in the selected folder(s). Choose which of these samples you actually want to analyze in case you are not interested in analyzing all the samples from a particular folder (figure 11.49).

8. In the third wizard window, you can see that the Result Metadata Table you specified earlier is already selected. Check that it is indeed the Result Metadata Table you intended to use and click Next.

9. The rest of the workflow is similar to the one described in section 11.3.2. Refer to this section to understand what parameters can be set, and which outputs are generated.

10. In the last Result Handling window, we recommend saving the batch results in separate folders.
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 14.1).
- 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 10.4.

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 is uploaded to the workbench when installing the Microbial Genomics Module. All variant track files must have the same reference genome for the Multi-VCF export to work.

### 11.4 Extract Regions from Tracks

The Extract Regions from Tracks tool facilitates specific extraction of mapped reads covering the particular regions specified by the annotation track, i.e., in this case the regions specified by an MLST scheme. The Extract Regions from Track tool is part of the template workflows Type a Known Species and Type among Multiple Species and 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:

![Microbial Genomics Module](Microbial Genomics Module) | Typing and Epidemiology (beta) | Extract Regions from Track
The input file to the tool is a Reads Track, Genome Track or and/or Annotation Track (figure 11.50) while the specification of the regions to be extracted is specified by the NGS MLST Annotation Track (figure 11.51) generated when initially running the Identify MLST tool.

Figure 11.50: The first input file to the Extract Regions from Tracks tool is a Reads Track, Genome Track or and/or Annotation Track.

Figure 11.51: The second input file to the Extract Regions from Tracks tool is a NGS-MLST Annotation Track generated by the Identify MLST tool.

The extracted mapped reads is visualized in the track list shown in figure 11.52. Through the navigation tool at top right, it is possible to switch among the various MLST regions defined for the analyzed species.

Figure 11.52: 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.
Chapter 12

Find the best matching reference

12.1 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. This section intends to describe the tool if you would like to use it as such. However, we recommend to use the Type a Known Species or Type among Multiple Species template workflows instead as described in chapter 11.

To identify best matching bacterial genome reference, go to:

Microbial Genomics Module ( ) | Typing and Epidemiology (beta) ( ) | Find Best Matches using K-mer Spectra ( )

Select the sequences you want want to find a best match sequence for (figure 12.1).

![Figure 12.1: To identify best matching reference, specification of read file is the first step.](image)

Select then a reference database, and specify the following settings (figure 12.2).

- **References** may be a single- or multiple list(s) of sequences. 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.
CHAPTER 12. FIND THE BEST MATCHING REFERENCE

Figure 12.2: Specify reference list 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 12.3).

Figure 12.3: Choose your output option before saving your results.

- **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.
  
  - **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 contaminations 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 10.5. **Note** that Best match results are added automatically to Result Metadata Table when using the template Type a Known Species and/or Type among Multiple Species workflow(s) or their customized versions.

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 12.4).

Figure 12.4: Taxonomy of the best matching reference and quality information is shown in the Metadata Result Table.

### 12.2 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 12.5).

![Figure 12.5](image-url)

Figure 12.5: 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.

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 14.2. 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.
Chapter 13

Resistance typing

Identification of antimicrobial resistance genes is important for understanding the underlying mechanisms and the epidemiology of antimicrobial resistance. The Find Resistance 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 http://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 11.

The Find Resistance tool is inspired by [Zankari et al., 2012] and uses BLAST for identification of acquired antimicrobial resistance genes within whole-genome sequencing (WGS) data.

13.1 Find Resistance

To perform resistance typing, go to:

Microbial Genomics Module ( ) Typing and Epidemiology (beta) ( ) Find Resistance ( )

Select the input genome or contigs (figure 13.1).

Figure 13.1: Pre-assembled and complete- or partial genomes simple contig sequences may be used as input for resistance typing.

You can then specify the settings for the tool (figure 13.2).

- DB Select the downloaded database for Find Resistance (using the Download of Database
CHAPTER 13. RESISTANCE TYPING

Figure 13.2: Select database and settings for resistance typing.

for Find Resistance tool (see section 17.1).

- **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 that agree on contig, source and predicted phenotype where one is contained by the other.

The output of the find Resistance 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), the location where the gene was found (contig name, and position in the contig) as well as a link to NCBI. To add the obtained resistance types to your Result Metadata Table, see section 10.5.
Chapter 14

Phylogenetic trees using SNPs or k-mers

14.1 Create SNP Tree

The Create SNP Tree tool is inspired by Kaas et al., 2014. There are two ways to initiate creation of a SNP tree: from the Result Metadata Table (see subsection 10.4) or by running the tool from the Toolbox. Note that you can only create a SNP tree if you have identified a common reference for the different stains you are trying to type, and used it for read mapping and variant calling for each of these samples.

To create a SNP tree from the Toolbox:

Microbial Genomics Module ( ) | Typing and Epidemiology (beta) ( ) | Create SNP Tree ( )

Select the relevant read mappings as shown in figure 14.1.

Figure 14.1: Select read mappings to be included in the SNP tree analysis.

Alternatively, select data recursively by right-clicking on the folder name and selecting Add folder contents (recursively) (figure 14.2), but remember to double check that files relevant for the downstream analysis are selected. An efficient alternative to these methods is to use the Quick filtering functionality from the Metadata Result Table to filter easily the data and initiate the SNP tree creation.

Select the variant tracks you want to use (figure 14.3). The variant tracks determine which positions to include in the SNP tree. The variant tracks need to have the same reference as the previously selected read mappings. Under normal circumstances you would select one variant.
track for each read mapping given in the input step, but that is not a requirement.

Figure 14.3: Select variant tracks and specify relevant parameters before generation of a SNP tree.

The following **Parameters** may be specified before setting up the algorithm for the construction of the SNP tree (see figure **14.3**):

- **SNV parameters**
  - **Variant tracks.** Select the variant tracks you want to use. The variant tracks determine which positions to include in the SNP tree.
  - **Include MNVs** or not, along with SNVs when building the SNP tree.
  - **Minimum coverage required in each sample** on a given position. The position is skipped if at least one sample has coverage below the specified threshold.
  - **Minimum coverage percentage of average required** on a given position. The position is skipped if at least one sample has coverage below this percentage of its own average coverage.
  - **Prune distance** specifies the 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 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.

- Result metadata
  - Result metadata Table. Specify location of the Result metadata table file.

- Tree view
  - Tree view settings. 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 Tree Settings in general: http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.

The variant calls and read mapping results are used to determine the SNP positions used in the tree. Note that the variant tracks are only used to determine which positions to include in the SNP tree. Only the position and the type (SNP, and MNV if enabled) are used, whereas any information about reference and allele is ignored. The read mappings are then used to estimate the consensus sequence. Only a variant with relative frequency above 50% (haploid organisms) will be effectively considered.

The initial list of variants is reduced as the following: All but one variant from the initial variant lists that fall within the specified pruning distance (for example 10nt) are ignored. Positions that are not well or not covered in one or more read mappings ("Minimum coverage required in each sample" and "Minimum coverage of average required") are removed. In addition, all SNPs which do not have the minimal z-score are excluded.

Select the tree construction algorithm you want to use (figure 14.4).

Figure 14.4: Choose the tree construction algorithm.

When selecting the Neighbor Joining method to create the tree, branch lengths are based on the distance between samples. The distance between two samples is computed as "Number of input positions used where the consensus sequence is different" / "Number of input positions used". The distance is therefore a number between 0 (no difference found in the input positions used) and 1 (all input positions used were different). From the tree, one can compute the distance between two samples by summing up all branches connecting them.

When selecting the Maximum Likelihood method to construct the phylogenetic tree, the next step of the wizard will be to specify the details of the evolutionary model to be used and to specify whether bootstrapping should be performed (see figure 14.5).

The first step of the maximum likelihood algorithm is to produce an alignment of the concatenated SNPs which is to be given as a starting point for the maximum likelihood phylogeny. The Create SNP Tree tool can optionally output this SNP alignment (see figure 14.6).

For CLC Genomics Workbench users, it may be beneficial to use the SNP alignment as input for the Model Testing tool (described in detail here http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Model_Testing.html). This tool can quantify which evolutionary model best suits the data best. You can then rerun the Create SNP Tree tool with the settings set as suggested by the Model Testing tool.
The maximum likelihood tool starts by creating a starting tree using the Neighbor Joining method. Then it proceeds to calculate the most likely phylogenetic tree under the given evolutionary model.

14.1.1 SNP tree output report

The Create SNP Tree tool can optionally output a report that summarizes the consequence of the applied filtering settings, as well as a summary of ignored positions attributed to the different read mappings.

The first section of the report, "Filter Status", contains the actual counts found during analysis. The different categories are described below:

- **Number of different input positions** corresponds to the number of unique positions based on SNPs (and MNVs if selected).
- **Pruned** count reflects the number of positions pruned based on the specified filter value (see above).
- **Coverage filtered** count reflects the number of positions filtered based on the two specified filter values (see above).
- **Z-value filtered** count reflects the number of positions filtered based on the specified filter value (see above).
- **Number of input positions used** the final number of positions that passed all filtering steps and used for creating the SNP tree.

The second section consists of a graphical and a tabular representation of the number of positions that was filtered because of the individual read mappings. The tabular view has three columns:

- **Read mapping**. The name of the read mapping.
- **Filtered, total**. The total number of filtered positions contributed by this read mapping, i.e. how many positions in this read mapping where either the coverage was too low, or the z-score was too low. **Note** that several read mappings can contribute the same position.
- **Filtered, only by this**. The number of filtered positions contributed only by this read mapping, i.e. the number of positions that was filtered by this read mapping, and not filtered by any other read mappings.

In the example below, we are creating a tree based on three read mappings, mapping 1, mapping 2 and mapping 3.

- Mapping 1 has low coverage on positions a, b and c.
- Mapping 2 has low coverage on positions a, b, d and e.
- Mapping 3 has low coverage on positions a, b and e.

In this case
• Mapping 1 will have a "Filtered, total" count of 3 (positions a, b and c), and a "Filtered, only by this" count of 1 (position c).

• Mapping 2 will have a "Filtered, total" count of 4 (positions a, b, d and e) and a "Filtered, only by this" count of 1 (position d).

• Mapping 3 will have a "Filtered, total" count of 3 (positions a, b and e) and a "Filtered, only by this" count of 0.

We will get a table like this:

<table>
<thead>
<tr>
<th>Read mapping</th>
<th>Filtered, total</th>
<th>Filtered, only by this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping 1</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Mapping 2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Mapping 3</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

An example of report content shown in figures 14.7, 14.8, 14.9 and 14.10.

Figure 14.7: Report on Filter Status for the created SNP tree.

Figure 14.8: Visualization of the filter effect across data used for generation of SNP tree.

In the applied example, ERR277235_1 shows significantly higher number of positions ignored than the other samples (see figures 14.8 and 14.10), and one might consider rerunning the tree without this sample in an attempt to get a higher resolution in the tree.
CHAPTER 14. PHYLOGENETIC TREES USING SNPS OR K-MERS

14.1.2 Visualization of SNP Tree including metadata and analysis result metadata

The overall tree layout, node- and label settings etc is defined by selected Tree Settings\(^1\) (figure 14.11).

The creation of a SNP tree allows for the associated Result Metadata Table to be specified. Thus it is possible to visualize the associated metadata as well as analysis result metadata on the phylogenetic tree. This makes it easier to find potential correlations across samples and verify, such as how well MLST typing and the tree topology fit together.

14.1.3 SNP Tree Variants editor viewer

It is possible to inspect the actual difference in nucleotide counts at a given internal node. This can be done by selecting the internal node of interest (figure 14.12) and then switching from the Show Tree element view to the Show SNP Tree Variants element view in the lower left corner of the visualization area.

Once in the SNP Tree Variant element view, the table lists the actual SNP input positions used for creation of the SNP tree. The number of rows (in figure 14.13 this is 62,556) corresponds to the

---

\(^1\) http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html
CHAPTER 14. PHYLOGENETIC TREES USING SNPS OR K-MERS

14.1.4 SNP Matrix

The Create SNP Tree tool can optionally output a matrix containing the pairwise number of SNP differences between all pairs of samples included in the analysis (see figure 14.14).
CHAPTER 14. PHYLOGENETIC TREES USING SNPS OR K-MERS 132

Figure 14.13: Counts of differences at a given position in the branches of the selected internal node.

Figure 14.14: A SNP matrix.

It is possible to easily get an overview of which samples are closely related by using the colored Comparison gradient in the side panel. The minimum and maximum values of the Comparison gradient scale can be adjusted by drag-and-drop such that the colors of the matrix cells change accordingly. Clicking on the Comparison gradients color scale a drop down menu appears where it is possible to select a different color scheme. Finally, it is possible to use the Lower threshold field to type in a 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. It is possible to export the SNP Matrix to any table format file (*.csv, *.xls, *.xlsx).

Note: it is not possible to import from a *.csv format to a SNP Matrix - rather export the SNP Matrix to a *.clc format and then import it.

14.2 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. There are two ways to initiate creation of a k-mer tree: either from the Result Metadata Table (see chapter 10.4), or from the Toolbox.

To run the Create K-mer Tree from the toolbox:

Microbial Genomics Module | Typing and Epidemiology (beta) | Create K-mer Tree

Input files can be specified step-by-step like shown in figure 14.15 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.

Figure 14.15: Selection of individual reads and single sequences or sequence list to be included in the K-mer tree analysis.

Specify the following parameters (figure 14.16):

- **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](https://en.wikipedia.org/wiki/Jaccard_index) and [https://en.wikipedia.org/wiki/Alignment-free_sequence_analysis](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: [http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html](http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html).

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.

### 14.2.1 Visualization of K-mer Tree for identification of common reference

The k-mer tree below (figure 14.17) includes 46 samples and 44 *Salmonella* genomes. To identify a candidate common reference genome, the tree was visualized using the radial tree topology.
Various parameters may be set before generation of a K-mer tree.

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 (accession NC_011083) located in the central region of the tree was 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 http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Tree_Settings.html.
Figure 14.17: 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.
Chapter 15

NGS MLST

In outbreak analysis, Multilocus Sequence Typing (MLST) is a central tool for identification of isolated pathogenic bacteria. Typing requires a defined MLST scheme for the targeted species which is based on a number of loci (typically 7). These fragments are usually chosen to lie in conserved regions such as housekeeping genes across all species members. For each locus, the scheme consists of a list of the known alleles. Each allele is assigned a number, and by using this numbering system each new isolate can be matched to known or new allelic profiles or sequence types (STs).

Usage of curated species-specific databases of MLSTs allows the epidemiologist to place the strain at hand in global and local outbreak histories. The PCR and Sanger sequencing steps are now routinely replaced by the more cost efficient and faster whole genome shotgun sequencing for which this tool is designed. In the CLC Microbial Genomics Module we offer a number of tools in connection to NGS-MLST:

- Download of existing MLST schemes from PubMLST, MLST.net or other sources.
- Creation of a new MLST scheme.
- Merge of available MLST schemes.
- Addition of sequence to MLST scheme.
- Identification of MLST based on read mapping.

In the CLC Microbial Genomics Module, an MLST scheme can be created in two ways:

- Downloaded from a database.
- Created from sequence data imported into the workbench.

The schemes downloaded from a database will already have information about many sequence types, whereas this information needs to be accumulated by the user for self-made schemes.

For NGS-MLST typing and summary of analysis results, we recommend using the Type a Known Species or Type among Multiple Species template workflows as described in chapter 11.
15.1 Schemes visualization and management

An MLST scheme consists of the following:

- Names of the loci
- For each of these loci a list of the known alleles
- A list of the known allelic profiles or sequence types described by the scheme

Given this information, sequence data from a new isolate can be used to assign a sequence type.

There are two ways of viewing a MLST scheme (besides the history view): once a MLST scheme is open, switching between the two table views of the scheme is done by clicking the buttons at the lower left corner of the view.

The profile table

The profile table shows all the sequence types in one table (see an example in figure 15.1).

![The profile table](image)

Figure 15.1: The profile table.

The first column in the table shows the Sequence Type (ST). In the succeeding columns, the allelic variants that were identified for each locus are reported. The Sequence Type is derived from the combination of these allelic types. In the example above (figure 15.1), the last column displays information about the **Clonal complex** of this sequence type. This information is not available in all schemes.

The table can be sorted by clicking the column headers. Pressing Ctrl (or ⌘ on Mac) while clicking the column makes a secondary sorting within the primary sorting.

From this table you can remove sequence types from the scheme:

- Select the sequence types you wish to remove | right-click | Remove Sequence Type

This will remove the sequence type from the scheme. The allele sequences will not be removed, since they can be shared among several sequence types. For information about adding sequence information to a scheme, see section 19.6.
CHAPTER 15. NGS MLST

The allele table

While the profile table described above provides an overview of the sequence types in a given scheme, the allele table provides an overview of the alleles that are included for each locus (see figure 15.2).

![Figure 15.2: The allele table.](image)

At the top of the view is a master table including all loci of the scheme (in the example shown in figure 15.2 there are seven loci). Selecting a row in the master table updates the allele table below which displays all the alleles of the selected locus. The columns in the allele table provide information about allele name, allele number, fragment length, creation data and a list of the sequence types in which a given allele is found.

From this table you can create an alignment of the sequences of two or more allelic types:

Select the allelic types | right-click | Align Sequences

This will show an alignment parameters dialog. To learn more about these parameters, click the dialog Help button.

You can also remove one of the allelic types:

Select the allelic type | right-click | Remove

Note! This will also remove all the sequence types containing this allelic type from the scheme. If you remove an allelic type by accident, you can click Undo.

15.2 Identify MLST

This section describes how to perform the identification of the MLST from a genome or a read mapping and given a MLST scheme. The Identify MLST tool requires an exact match between the allele sequences in the MLST scheme and a consensus sequence. If a read mapping is used as
input, a consensus of the read mapping will be used. This stringency level is needed because the alleles used for multi locus sequence typing are very similar, often almost identical. Therefore MLST typing relies on a MLST scheme that represent the organism under investigation well.

To MLST type a consensus sequence or a read mapping, go to:

Microbial Genomics Module (糌) | Typing and Epidemiology (beta) (糌) | NGS-MLST (糌) | Identify MLST (糌)

In the first wizard window you are asked to select the consensus sequence or read mapping to be used for MLST typing (figure 15.3).

![Figure 15.3: Select consensus sequence or read mapping to be used for MLST typing.](image)

In the next window, two options should be specified (figure 15.4):

![Figure 15.4: Select relevant MLST scheme(s) for typing.](image)

- **Scheme** The NGS-MLST typing tool requires defined fragments of multiple housekeeping genes in the format of a MLST scheme, which should specified as a parameter (figure 15.4).

- **Low coverage reported when below** enable the user to specify the minimum coverage threshold. If the coverage within a found loci falls below this threshold, a warning will be printed in the output report.

The tool generates two output files: the **Consensus NGS MLST Report** and the **NGS MLST Annotation Track**. The **Consensus NGS MLST Report** includes information on which MLST
scheme was applied, the identified sequence type (ST) as well as an overview table which summarizes the targeted genes, their consensus sequence profile and coverage. The NGS MLST Annotation Track specifies the individual target regions in the coordinate system of the original input sequence.

In situations where not all (potentially no) loci are found in the consensus sequence, and a loci positions is transferable from the read mapping reference sequence, it will be reported as a possible new loci at the transferred position.

To add the obtained ST to your Result Metadata Table, see section 10.5. Note that results are added automatically to the Result Metadata Table when using the templates Type a Known Species and Type among Multiple Species workflows or their customized versions.

### 15.3 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 step is preliminary to the Identify MLST tool in case you are working with a sample containing a single or multiple unknown species, as in the Type among Multiple Species workflow. You will input a sequence, or a sequence list (figure 15.5).

![Figure 15.5: Select relevant genome sequence or sequence list.](image)

Then select as many MLST schemes are necessary to identify the species present in your sample (figure 15.6).

![Figure 15.6: Select relevant MLST scheme(s) to search among.](image)

The output of this tool is a MLST scheme that matches best the sequence(s) analyzed. To add the obtained best match to a Result Metadata Table, see section 10.5. Note that results are added automatically to the Result Metadata Table when using the template Type among Multiple Species workflow.
Part IV

Databases
Chapter 16

Databases for Taxonomic Profiling

16.1 Create Microbial Reference Database

The Create Microbial Reference Database tool downloads selected references from GenBank and RefSeq, and outputs a single sequence list with all the necessary annotations for the taxonomic profiling (i.e., assembly IDs).

To run the tool, go to:

Microbial Genomics Module | Databases | Taxonomic Analyses | Create Microbial Reference Database

In the first window (figure 16.1), select the source of the database you wish to generate.

Figure 16.1: Select the references you want to download.

You can choose to download a curated database. You can choose the full database, or one optimized to contain only contig with a length of at least 250,000 bp. This optimized small-size reference database is particularly useful for running the Taxonomic Profiling tool on a laptop computer.

If you select "Create custom reference database", click Next to select the source of the database you wish to generate (as in figure 16.2)

You can choose from:

- Prokaryotes: Bacteria and/or Archaea
Figure 16.2: Choose the type of reference you want for your custom database.

- Eukaryotes: Fungi and/or Protozoa
- Virus. Note that downloading choosing this option will result in both virus and bacterial assemblies. Indeed, viruses are identified according to their BioProject ID, but this ID also refers to bacterial assemblies that were sequenced together with the virus. Filtering the table on taxonomy will allow you to only see viruses.

You can also use other sources:

- Provide a list of Genbank accession numbers in the white field, or
- Browse your computer for a file with accession numbers, or
- Browse the Navigation Area of the workbench for a sequence list. The corresponding references will be appended to the downloaded sequence list automatically.

The time it will take to download the data (such as assembly summaries, genome report) depends on how many databases are downloaded and the bandwidth of your internet connection. No sequence data is downloaded at this point.

The tool will open a table called a Database builder (figure 16.3) from which you can design your own database. A series of functionality can help you filter and sort the table to extract the information relevant to your project.

1. Use the "Quick selection" button to quickly select predefined subsets for download:
   - Single scaffold complete genomes in RefSeq
   - Complete genomes in RefSeq
CHAPTER 16. DATABASES FOR TAXONOMIC PROFILING

Figure 16.3: Output table from the Create Microbial Reference Database tool.

- All complete genomes

Each reference in the table is marked with one of the following statuses: Complete genome, Chromosome, Scaffold or Contig. In addition to this some references are marked as representative genomes for a clade (repr) or as reference genomes (refr). We include references that are marked as Complete genome, Chromosome, representative genome and/or reference genome in these subsets.

2. Aggregate the table to a specified taxonomic group using the drop down menu in the "Data" palette of the side panel. Use the category "Name" to de-aggregate the table.

3. Use filter(s) to keep only the rows you are interested in, and click on the button "Include all" to create a database with the remaining rows.

4. Alternatively, click on "Include all" rows first, set one or several filters, and use the button "Exclude all" on the remaining rows. Clear the filter(s) by clicking on the red buttons next to each filter set. The rows not filtered away in the second step should still be checked.

Once the table contains all desired rows, click Download selection. Close to the button, you can check how many references are selected, and an estimate of the total size of the selection.

The dialog shown in figure 16.4 allows you to set an additional filter "Minimum contig length" (except if you have selected the curated database). It also warns about the memory and disk requirements that will be needed to later run the Taxonomic Profiling tool with the database you are about to download.

16.2 Set Up Microbial Reference Database

The Set Up Microbial Reference Database tool adds metadata to individual sequences in a sequence list so microbial isolate samples can be analyzed in the context of this metadata
for typing. In general, the tool will be used to associate a user’s own metadata to reference genomes included in a pathogen reference genome database (such as the one downloaded by the Download Microbial Reference Genomes tool).

The tool import pathogen information to an existing database and bundle the new metadata or overwrite the existing metadata reference by reference if this option is selected. This means that references that already have metadata and for which no new metadata is imported will keep the metadata; references with metadata for which new metadata is imported will be updated; references with no metadata will acquire new metadata.

To run the tool, go to:

Microbial Genomics Module | Databases | Taxonomic Analyses | Set Up Microbial Reference Database

In the first window, choose a new sequence list or an existing database (figure 16.5).

In the second dialog (figure 16.6), select an Excel spreadsheet or a CSV file saved on your computer.

The input file must have a Name column to ensure that the link between the sequence and the metadata is successful. In addition, it is highly recommended - but optional - to have a Taxonomy column for use with other downstream tools (especially in cases where the sequence list does not have this information already). Taxonomy must be following the Qiime or common 7-level formats.

- Qiime format, such as "k_Bacteria, p_Proteobacteria, c_Gammaproteobacteria, o_Enterobacterales,"
f. Enterobacteriaceae, g. Klebsiella, s. Klebsiella pneumoniae"

- The common 7-level, semi-colon separated format such as "Bacteria; Proteobacteria; Gammaproteobacteria; Enterobacteriales; Enterobacteriaceae; Klebsiella; Klebsiella pneumoniae", going from (super)kingdom to species.

You can add as many columns of metadata as you wish. The names given in the first row will be used as metadata categories. Once imported, the information from the spreadsheet (or CSV file) will fill in the table included in the lower part of the wizard window (see figure 16.4), and the headers will take the names of the first row. It is still possible to edit the first row data at this point, thereby changing the names of the metadata categories. Leaving a first row field blank means that the metadata in that column will not be imported.

The tool will output a single sequence list containing both sequence data and metadata for each sequence initially present in the input sequence list. By selecting the "Overwrite old metadata" option, the references already associated with metadata for which new metadata is imported will be updated.

In the last wizard window, the option to generate a report is selected by default. The report contains the following summary information:

- Number of sequences in the input sequence list, and in the output sequence list
- Number of sequences skipped because of duplicated names
- Sequences left unchanged and sequences for which metadata was updated
- Number of metadata columns in the input metadata table
16.3 Download Pathogen Reference Database

Download a collection of bacterial genomes representatives directly from RefSeq or NCBI Pathogen Detection Project (see http://www.ncbi.nlm.nih.gov/projects/pathogens/).

Microbial Genomics Module | Databases | Taxonomic Analyses | Download Pathogen Reference Database

This will open the following wizard window (figure 16.7):

![Figure 16.7: Downloading from the NCBI's RefSeq bacterial genomes database.](image)

The settings are:

- **Select download source** RefSeq or Pathogen Project.
- **Select the organism you want to download.** This can be done By Kingdom/Domain, in which case you can choose a set from the predefined list (all archa, all bacteria, all fungi, all microbes, all plasmids, all viruses) or **Pathogens**, in which case you can choose an available set of microbes.
• **Select a filter for the selected species**: It is possible to **filter before downloading on organism name**, thereby potentially saving a lot of time for the download, or **filter after download on name and description** by defining one or more phrases to limit the output to those sequences which include at least one of these phrases in their description or their name.

• **Only complete genomes**

• **Minimum Length**: is by default set to 500,000 nt to favor bacterial genomes instead of plasmids.

• **Minimum N50 length** (the default value is set at 1,000,000 bp)

• **Maximum number of contigs** (the default value is set at 100) to ensure high enough quality of the database content

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 imported database includes a list of different bacterial genome sequences as well as the associated accession numbers (acc nos), 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

### 16.3.1 Extracting a subset of a database

After download, it is always possible to select a subset of bacterial genomes and saving the reduced list in a separate file. This can reduce significantly subsequent analysis runtime.

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 16.8).

4. **Select** all remaining rows.

5. **Click** the *Create New Sequence List* button.

6. **Save** the subset reference list.

### 16.3.2 Download Pathogen Reference Database output report

You can choose to generate a report with the following summary data for each downloaded database:

- Number of reference genomes before filtering
- Number of reference genomes after filtering
- Number of output sequences
- Database version number
- Date of release of the database on the NCBI Pathogen Detection project’s ftp site
- Sequence metadata statistics
- Sequence taxonomy statistics, including the amount of entries without taxonomy information.

It is possible to edit the missing or incorrect taxonomic entries in the sequence list table manually by right-clicking on the field to be added or edited, or by using the *Set Up Microbial Reference Database* tool (see section 16.2).
It is also possible to merge multiple reference genome databases to add genomes of interest that are not present in the online database, as well as to delete a reference genome from a pathogen reference genome database when references are not relevant to the analysis or when they are of poor quality.

The date of each full database download, as well as the ftp address of the root of the source NCBI Pathogen Detection Project database will be included in the log of the tool and in the history of the downloaded files such that users can locate the downloaded files on NCBI’s ftp site again at a later time.
Chapter 17

Databases for Functional Analysis

17.1 Download Database for Find Resistance

The Download Database for Find Resistance tool automatically downloads the database from https://cge.cbs.dtu.dk/services/data.php. To run the tool, go to:

Databases | Functional Analysis | Download Database for Find Resistance

Specify a location to save the database. We recommend to use a folder where you save all the databases and MLST schemes necessary to run the CLC Microbial Genomics Module tools. Click Finish.

The output is a sequence list that can be visualized as a table as well. In the table view, users can review the metadata associated with the sequences. It is possible to edit all metadata values, or to delete a gene in a resistance gene database such that users can keep their database up-to-date within the workbench.

It is also possible to merge two resistance gene databases, for example to merge a custom database with the one obtained using the Download Database for Find Resistance tool, or to add new genes to an existing database.

17.2 Set Up Gene Database

The Set Up Gene Database tool is meant for users who want to use their own list of resistance genes for resistance detection with the Find Resistance tool.

The tool can create new resistance gene lists/databases from a sequence list, or it can also be used to edit existing databases downloaded with the Download Database for Find Resistance tool. In the latter case, the tool imports resistance information to an existing resistance gene database and bundles the new metadata or overwrites the existing metadata sequence by sequence if this option is selected. This means that sequences that already have metadata and for which no new metadata is imported will keep the metadata; sequences with metadata for which new metadata is imported will be updated; sequences with no metadata will acquire new metadata.

To run the tool, go to:

Databases | Functional Analysis | Set Up Gene Database
In the first window (figure 17.1), select a new sequence list or an existing database (with or without resistance genes information).

Figure 17.1: Select a sequence list or an existing resistance database you wish to modify.

In the second window (figure 17.2), select an Excel file (or a CSV file) containing the resistance information previously saved on your computer.

Figure 17.2: Select the file in which you saved the information you want to add to the sequence list.

There are 2 required columns in the input file: "Name" and "Phenotype" (i.e., resistance). You can add as many columns of metadata as necessary. The names given in the first row will be used as metadata categories.

Once imported, the information from the spreadsheet or CSV file will fill in the table included in the wizard window (figure 17.2), and the headers will take the names of the first row. It is still
possible to edit the first row data at this point, thereby changing the names of the metadata categories. Leaving a first row field blank means that the metadata in that column will not be imported.

The required columns "Name" and "Phenotype", as well as the optional columns "Function" and "Accession number" from Resfinder are highlighted in yellow as they hold the information used subsequently by the Find Resistance tool.

Note that the option to overwrite previous metadata is selected by default but you can choose to deselect it in this window.

The tool will output a new database (or an updated database) containing the data included in the spreadsheet for all sequences whose name matched with the ones specified in the "Name" column of the spreadsheet. It is also possible to generate a report containing the number of genes in the database and the list of resistance phenotypes in the database (figure 17.3).

1 Set Up Resistance Gene Database

<table>
<thead>
<tr>
<th>Key</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Genes input</td>
<td>4</td>
</tr>
<tr>
<td>Genes skipped because of duplicate names</td>
<td>0</td>
</tr>
<tr>
<td>Genes skipped because of missing metadata match</td>
<td>0</td>
</tr>
<tr>
<td>Distinct resistance phenotypes</td>
<td>3</td>
</tr>
</tbody>
</table>

List of phenotypes in database

- Resistance A
- Resistance C
- Resistance D

Figure 17.3: Example of a report produced by the Set Up Gene Database tool.

17.3 Download Protein Database

The Download Protein Database allows you to download select versions of the following protein databases:

- Clusters of Orthologous Genes (COG)
- SwissPROT
- UniProt Reference Clusters (UniRef50)
- UniProt Reference Clusters (UniRef90)
- UniProt Reference Clusters (UniRef100)

These Protein databases are used to convert CDS annotation by the Annotate CDS with Best DIAMOND Hit tool (see section 7.3).

To run the tool, go to:

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. The output of this tool is a new sequence list that can be used as input to the Annotate CDS with Best DIAMOND Hit tool.
17.4 Download GO Database

The Download GO Database allows you to download a current copy of the GO database and of Pfam 2 GO mappings from the Gene Ontology Consortium (http://geneontology.org/). The GO database is used to convert Pfam annotation to GO terms by the Annotate CDS with Pfam Domains tool (see section 7.4) and by the Build Functional Profile tool (see section 7.5).

To run the tool, go to:

Databases | Functional Analysis | Download GO Database

The tool creates a new object called GO database in the Navigation Area. 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 it was downloaded, and the number of entries in the file.
Chapter 18

Databases for Amplicon-Based Analysis

18.1 Download Amplicon-Based Reference Database

OTU reference databases contain representative OTU sequences and their taxonomy. They are needed to perform reference-based OTU clustering. Three popular reference OTU databases, clustered at various similarity percentages, can be downloaded using the Download Amplicon-Based Reference Database tool:

- **Greengenes**: 16S rRNA gene from ribosomal Small Subunit for Prokaryotic taxonomic assignment clustered OTUs at different percentages. [http://greengenes.secondgenome.com/downloads](http://greengenes.secondgenome.com/downloads)
- **Silva/Arb SSU**: 16S/18S rRNA from ribosomal Small Subunit for Prokaryotic and Eukaryotic taxonomic assignment clustered OTUs at different percentages. [http://www.arb-silva.de/download/archive/qiime/](http://www.arb-silva.de/download/archive/qiime/)
- **UNITE**: ITS spacer clustered OTUs at different percentages for fungal taxonomic assignment. [https://unite.ut.ee/repository.php](https://unite.ut.ee/repository.php)

To run the tool, go to

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 look at the databases in a table format, you can see that only the columns Name, Size and Start of Sequence are populated, leaving the column about taxonomy empty. That is because taxonomy for each sequence is embedded in the sequence object used to create OTUs, while it will not appear in the database tables. As none of the tools will ever use the information present in the "Taxonomy" column from these tables, it is irrelevant to try to add or edit the names in them yourself.

If you wish to format your own database with your own QIIME formatted sequence and a corresponding taxonomy file, use the tool called Set Up Amplicon-Based Reference Database.
18.2 Set Up Amplicon-Based Reference Database

In addition to the download of reference databases using the Download Amplicon-Based Reference Database tool, you can format your own databases by running the Set Up Amplicon-Based Reference Database tool:

**Databases** | **Amplicon-Based Analysis** | **Set Up Amplicon-Based Reference Database**.

This tool takes as input QIIME-format fasta files (typically clustered into OTUs at 99, 97, 94, and 90 percent similarity) and corresponding taxonomy mapping files, where each line of the taxonomy file contains an OTU name and its taxonomy. The OTU name and its taxonomy must be separated by a tab character while taxonomy levels are separated with semicolons. For example, the following line

```
o123  k__Bacteria;p__Bacteroidetes;c__Sphingobacteria;o__;f__;g__;s__.
```

indicates that the OTU o123 belongs to the class Sphingobacteria and that its taxonomy is specified only up to the class level. For more information about QIIME format, see [http://qiime.org/index.html](http://qiime.org/index.html).
Chapter 19

Databases for NGS-MLST

19.1 Download MLST Schemes

For the most common bacteria, a MLST scheme has already been defined and can be downloaded from the databases at PubMLST.org.

To download the available MLST schemes, go to:

Microbial Genomics Module | Databases | NGS-MLST | Download MLST Schemes (PubMLST)

This will bring up a dialog as shown in figure 19.1.

![Figure 19.1: Selecting all MLST schemes for download.](image)

The list of is dynamically updated with the available MLST schemes. This means that it takes a short while before the dialog is shown (depending on your internet connection). You can select one or more schemes for download. Use the filter field to type the name of the schemes you are interested in.

We recommend to Save the schemes in the folder where you also saved the different databases you need to download for the CLC Microbial Genomics Module.

See section 15.1 for information about the content of the downloaded scheme(s).
19.2 Download other MLST Schemes

In addition to the MLST schemes that CLC Microbial Genomics Module can automatically download from PubMLST, you can also provide a URL to an MLST scheme located on another web server (see figure 19.2). It requires, however, that the scheme has the same format as the schemes on PubMLST.

![Figure 19.2: Provide a URL for downloading a custom scheme.](image)

To download other MLST databases, go to:

Microbial Genomics Module | Databases | NGS-MLST | Download other MLST schemes

We recommend to Save the schemes in the folder where you also saved the different databases you need to download for the CLC Microbial Genomics Module.

19.3 Create MLST Schemes

There are several ways to create a new MLST scheme: it is possible to create one using the Create MLST Schemes tool as explained in this section; it is also possible to add an NGS MLST report (generated using the Identify MLST tool) to a Scheme (see section 19.6). Finally, it is possible to use the CLC MLST module to run an assembly and create an isolate that can later be added to a scheme (see [http://resources.qiagenbioinformatics.com/manuals/clcmlst/current/CLC_MLST_Module_User_Manual.pdf](http://resources.qiagenbioinformatics.com/manuals/clcmlst/current/CLC_MLST_Module_User_Manual.pdf)).

A scheme can be created from scratch by defining a number of loci and accumulating sequence information about each loci. The schemes should have the same format as the schemes that are downloaded.

To create a new scheme, go to:

Microbial Genomics Module | Databases | NGS-MLST | Create MLST Scheme

This will bring up a dialog as shown in figure 19.3.

In the top field you can enter a name for the scheme - typically the name of the species in question. Below, you can specify the location for a text file where all the loci included in the scheme are listed. If such a file does not exist, you can just type, in the "Add genes to scheme" section of the dialog, the name of the locus in the Name of gene field, and click the Add button. If you accidently add a wrong name to the list, select it and click the Remove button. When you are done adding loci for your scheme, click Next to start assigning sequences to the loci.

The wizard will open successively one window for each locus in the scheme (figure 19.4). In each of these windows, you can assign allelic sequences to each locus. Clicking Next will display a
similar dialog for the next gene, and so on for all the loci that were entered in the first dialog.

It is not mandatory to add sequences to the loci. You can also choose to proceed and create the scheme before you add sequences. If you just add individual sequences as shown in figure 19.4, they will not be combined in an allelic profile and the scheme will not yet contain any sequence types.

19.4 Add NGS MLST Report to Scheme

Once the MLST typing has been performed, and new types have been identified, it is possible to use the generated Consensus NGS MLST Report as input for the Add NGS MLST Report to Scheme tool and hereby add new typing information to an existing MLST scheme.

Specify the report output file(s) of the Identify MLST tool to be used (figure 19.5).

Then select the scheme you would like to add the newly found MLST type(s) to (figure 19.6). This scheme should have the same genes listed than in the report(s) specified in the first step. Any report that have a mention "Not found" for the status and allelic profile will be considered as incompatible with a scheme. A warning is issued in the dialog and the tool cannot be started.

Save the updated scheme in the folder where you have stored database and downloaded
19.5 Merge MLST Schemes

If the database has updated a particular scheme (e.g., new alleles and/or sequence types), your local copy of the scheme also needs to be extended. This is done by merging the two schemes into a single scheme, as downloading of the new scheme version from the database would mean discarding your own contributions to the local copy of the scheme.

First, download the scheme from the updated database. Then go to:

Microbial Genomics Module | Databases | NGS-MLST | Merge MLST Schemes

This brings up a dialog where the two schemes that you wish to merge should be selected (figure 19.7). However, in order to merge, the schemes must have the same loci definitions (i.e., the same number of loci with identical names).

The newly merged scheme will open and can be saved. You can then safely delete the previous versions of the scheme in the Navigation Area.

Note that if there are duplicates among the sequence types in the scheme, only one of the types
will be included in the new scheme.

19.6 Add Sequences to MLST Schemes

MLST schemes can be extended by adding individual sequences to the alleles in the scheme:

Microbial Genomics Module | Databases | NGS-MLST | Add Sequences to MLST Scheme

This will bring up a dialog as shown in figure 19.8, where it is possible to select the scheme you would like to add gene(s) to.

In the next step(s) - there are as many wizard windows as genes in the scheme - it is possible to assign sequences to each gene successively (see figure 19.9).

Specify a location to save the scheme(s). We recommend to use a folder for saving all the databases and MLST schemes necessary to run the CLC Microbial Genomics Module tools.

The sequences added in this way cannot be combined to an allelic profile and a sequence type, and will thus only contribute to expanding the number of allelic types.
Part V

Panel Support
Chapter 20

QIAseq 16S/ITS Demultiplexer

The Panel Support section offers a new 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 20.1) or from a custom list.

If you choose to use a table of custom barcodes (figure 20.2), 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.
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 20.3), 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

<table>
<thead>
<tr>
<th>Region</th>
<th>Barcode</th>
<th>Number of reads</th>
<th>Percentage of reads</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y1V2</td>
<td>AGRGTTTATMNGGCTC-CTGCTGCTYCTGTA</td>
<td>39,381</td>
<td>7%</td>
</tr>
<tr>
<td>Y2V3</td>
<td>GCSCNNACGGGTTGATTA-WITTACCGCGCTGCTG</td>
<td>124,486</td>
<td>23%</td>
</tr>
<tr>
<td>Y3V4</td>
<td>CCTACCGGNNNGGGCAGC-CACTACHWGGGTTACATCC</td>
<td>92,961</td>
<td>17%</td>
</tr>
<tr>
<td>Y4V5</td>
<td>GTYCTAGCMCGCGCGTAA-CGGYCJATTYMWTAGT</td>
<td>61,912</td>
<td>11%</td>
</tr>
<tr>
<td>Y5V7</td>
<td>GAATTAGATACCCBGRCTGTC-ACGTOFRCCCGCGCCTCCTC</td>
<td>119,151</td>
<td>22%</td>
</tr>
<tr>
<td>Y7V9</td>
<td>YACCCAGCMMMACCC-TAGGGYTACCTTGTTAYGACTT</td>
<td>78,053</td>
<td>14%</td>
</tr>
<tr>
<td>ITS</td>
<td>CTGGGTATAGGGAAGCTAA-GCTGCTTCTCTCATGATG</td>
<td>7,028</td>
<td>1%</td>
</tr>
<tr>
<td>Ugrouped</td>
<td></td>
<td>31,479</td>
<td>6%</td>
</tr>
</tbody>
</table>

1.2 Reads per region: Barplot

Figure 20.3: An example of demultiplexing report.
Part VI

Appendices
Chapter 21

Legacy tools

A folder with the name Legacy Tools is found at the bottom of the toolbox. The tools found in this folder are tools that have been used in older versions of the Module but now have been replaced with new tools. The tools in the Legacy Tools folder will no longer be updated and will be retired in future versions of the Module. If possible, we recommend that you base your analysis on the new tools rather than using the legacy tools.

21.1 Optional Merge Paired Reads

The Optional Merge Paired Reads tool has been moved to Legacy Tools folder because it is now incorporated into the OTU Clustering tool. When the input consists of paired reads, the OTU clustering tool will initially merge them into pairs, and align the resulting paired reads to OTUs. Reads that cannot be merged will be independently aligned to reference OTUs. Then both reads of a pair will be assigned to the 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 ACGACGACGnnnnnnnnnnnnnnnnnnnnTACTACTAC.

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 using the Optional Merge Paired Reads tool. The Optional Merge Paired Reads tool will merge paired-end reads 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.

To run the Optional Merge Paired Reads tool, go to Microbial Genomics Module | Metagenomics | Amplicon-Based Analysis | Optional Merge Paired Reads.

Select any number of sequences as input. The tool accepts both paired and unpaired reads but will only merge the paired reads while returning the unpaired ones as "not merged" reads in the output. Note that paired reads have to be in forward-reverse orientation. After merging, the merged reads will always be in the forward orientation.
In order to understand how these parameters should be set, an explanation of the merging algorithm is needed: Because the fragment size is not an exact number of base pairs and is different from fragment to fragment, an alignment of the two reads has to be performed. If the alignment is *good and long enough*, the reads will be merged. *Good enough* in this context means that the alignment has to satisfy some user-specified score criteria (details below). Because of sequencing errors that typically are more abundant towards the end of the read, the alignment is not expected always to be perfect, and the user can decide how many errors are acceptable. *Long enough* in this context means that the overlap between the reads has to be non-coincidental. Merging two reads that do not really overlap leads to errors in the downstream analysis, thus it is very important to make sure that the overlap is big enough. If only a few bases overlap was required, some read pairs will match by chance, so this has to be avoided.

The following parameters are used to define what is *good enough* and *long enough*.

- **Mismatch cost**: The alignment awards one point for a match, and the mismatch cost is set by this parameter. The default value is 2.

- **Minimum score**: This is the minimum score of an alignment to be accepted for merging. The default value is 8. As an example: with default settings, this means that an overlap of 11 bases with one mismatch will be accepted (10 matches minus 2 for a mismatch).

- **Gap cost**: This is the cost for introducing an insertion or deletion in the alignment. The default value is 3.

- **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. However, this should be used with great care which is why the default value is 0. As explained above, a wrong decision to merge the reads leads to errors in the downstream analysis, so it is better to be conservative and accept fewer merged reads in the result. 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 main result will be two sequence lists for each sample selected as input to the tool: one containing the merged reads (labeled as "merged"), and one containing the reads that could not
be merged (labeled as "not merged"). Note that low quality can be one of the reasons why a pair cannot be merged. Hence, the list of reads that could not be paired is more likely to contain more reads with errors than the one with the merged reads.

### 21.2 Fixed Length Trimming

The Fixed Length Trimming tool has been moved to Legacy Tools folder because the OTU Clustering tool can now deal with reads with variable lengths. The input sequences are aligned to reference OTU sequences (e.g. the reference database) and, if the input sequence is shorter, the unaligned ends of the reference are ignored.

In previous versions of the software, in order to compare sequences and cluster them, they all needed to be of exact same length. All reads which are shorter than the cut-off were discarded, and reads longer than that were trimmed back to the chosen length.

To run the tool, go to

**Toolbox | Legacy tools | Fixed Length Trimming**

and select the sequences you would like to trim.

In the next wizard window you can enter manually the desired length for the trimmed reads. Alternatively, the **Fixed length Trimming** algorithm can calculate the trimming cut-off value as the mean length of the merged reads minus one standard deviation. If this option is chosen, it is important that all samples are trimmed at the same time as the mean and standard deviation for the combined reads in all samples needs to be estimated at once.

You can also offset one adapter or barcode by typing the nucleotide sequence in the Primer offset window. Exact matching is performed but ambiguous symbols are allowed. For more than one adapter, we recommend to perform prior to the Fixed Length trimming a **Trim Reads** step with Automatic read-through adapter trimming. are allowed. For more than one adapter, we recommend to perform prior to the Fixed Length trimming a **Trim Reads** step with Automatic read-through adapter trimming.
Chapter 22

Licensing requirements for the CLC Microbial Genomics Module

CLC Microbial Genomics Module can be installed in viewing mode in a workbench to access data that was created using the module. But in order to use the tools of CLC Microbial Genomics Module, you will need to have a license. This chapter describes in detail the different types of license available for workbench and server users.

22.1 Workbench licenses

You can manage the licenses for the CLC Microbial Genomics Module from the Plugins manager, by clicking on the “Import a new License” button. It will open the dialog seen in figure 22.1. The different options are explained in detail in the following sections.

Figure 22.1: Import a new License.
22.1.1 Request an evaluation license

We offer a fully functional version of the CLC Microbial Genomics Module for evaluation purposes, free of charge. Each user is entitled to 14 days demo of the workbench. If you are unable to complete your assessment in the available time, please send an email to bioinformatics-sales@qiagen.com to request an additional evaluation period.

When you choose the option **Request an evaluation license**, you will see the dialog shown in figure 22.2.

![Figure 22.2: Choosing between direct download or going to the license download web page.](image)

In this dialog, there are two options:

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

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

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

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

**Direct download**

After choosing the **Direct Download** option and clicking on the button labeled **Next**, the dialog shown in figure 22.3 appears.

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

**Go to license download web page**

After choosing the **Go to license download web page** option and clicking on the button labeled **Next**, the license download web page appears in a browser window, as shown in 22.4.

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

Back in the Workbench window, you will now see the dialog shown in 22.5.
Click the **Choose License File** button and browse to find the license file you saved. When you have selected the file, click on the button labeled **Next**.

### 22.1.2 Download a license using a license order ID

Using a license order ID, you can download a license file via the Workbench or using an online form. When you have chosen this option and clicked **Next** button, you will see the dialog shown in **22.6.** Enter your license order ID into the text field under the title License Order-ID. (The ID can be pasted into the box after copying it and then using menus or key combinations like Ctrl+V on some system or ⌘+V on Mac).

In this dialog, there are two options:

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

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

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

**Direct download**

After choosing the **Direct Download** option and clicking on the button labeled **Next**, the dialog shown in figure 22.7 appears.

![Direct Download dialog](image)

**Figure 22.7: A license has been downloaded.**

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

**Go to license download web page**

After choosing the **Go to license download web page** option and clicking on the button labeled **Next**, the license download web page appears in a browser window, as shown in 22.8.

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

Back in the Workbench window, you will now see the dialog shown in 22.9.
22.1.3 Import a license from a file

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

When you have clicked on the **Next** button, you will see the dialog shown in 22.10.

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

![License download web page](image)

Figure 22.8: The license download web page.

![Import license file](image)

Figure 22.9: Importing the license file downloaded from the web page.

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

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

To configure the Workbench to connect to a CLC License Server, select the **Configure License Server connection** option and click on the **Next** button. A dialog for the license server connection configuration is then presented. See figure 22.11.

![Figure 22.11: Connecting to a CLC License Server.](image)

The options in that dialog are:

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

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

- **Manually specify license server.** Select this option to enter the details of the machine the CLC License Server software is running on, specifically:
  
  - **Host name.** The address for the machine the CLC License Server software is running on.
  
  - **Port.** The port used by the CLC License Server to receive requests.

- **Use custom username when requesting a license.** Optional. If this is checked, a username can be entered. That will be passed to the CLC License Server instead of the username of the account being used to run the Workbench.

- **Disable license borrowing on this computer.** Check this box if you do not want users of the computer to borrow a license. See section 22.1.4 for further details.
### Special note on modules needing a license

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

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

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

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

### Borrowing a license

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

The procedure for borrowing a license is:

1. Go to the Workbench menu option:
   
   **Help | License Manager**

2. Click on the "Borrow License" tab to display the dialog shown in figure 22.12.

   ![Figure 22.12: Borrow a license.](image)

3. Use the checkboxes at the right hand side of the table in the License overview section of the window to select the license(s) that you wish to borrow.

4. Select the length of time you wish to borrow the license(s).

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

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

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

**Common issues when using a network license**

**No license available at the moment**  If all the licenses are in use, you will see a dialog like that shown in figure 22.13 when you start up the Workbench.

![Figure 22.13: This window appears when there are no available network licenses for the software you are running.](image)

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

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

**Lost connection to the CLC License Server**  If the Workbench connection to the CLC License Server is lost, you will see a dialog as shown in figure 22.14.

![Figure 22.14: This message appears if the Workbench is unable to establish a connection to a CLC License server.](image)
If you have chosen the option to **Automatically detect license server** and you have not succeeded in connecting to the License Server before, please check with your local IT support that automatic detection will be possible to do at your site. If it is not possible at your site, you will need to manually configure the CLC License Server settings using the License Manager, as described earlier in this section.

If you have successfully contacted the CLC License Server from your Workbench previously, please consider discussing this issue with your CLC License Server administrator or your local IT support, to make sure that the CLC License Server is running and that your Workbench can connect to it. There may be situations where you wish to use a different license or view information about the license(s) the Workbench is currently using. To do this, open the License Manager using the menu option:

**Help | License Manager**

The license manager is shown in figure 22.15.

This dialog can be used to:

- See information about the license (e.g. what kind of license, when it expires)
- Configure how to connect to a license server (**Configure License Server** the button at the lower left corner). Clicking this button will display a dialog similar to figure 22.11.
- Upgrade from an evaluation license by clicking the **Upgrade license** button. This will display the dialog shown in figure 2.3.
- Export license information to a text file.
- Borrow a license
If you wish to switch away from using a network license, click on the button to **Configure License Server** and uncheck the box beside the text **Enable license server connection** in the dialog. When you restart the Workbench, you can set up the new license as described in section 2.3.

### 22.1.5 Download a static license on a non-networked computer

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

- Install the CLC Microbial Genomics Module on the machine you wish to run the software on.
- Start up the software as an administrative user and find the host ID of the machine that you will run the CLC Workbench on. You can see the host ID the machine reported at the bottom of the License Manager window in grey text.
- Make a copy of this host ID such that you can use it on a machine that has internet access.
- Go to a computer with internet access, open a browser window and go to the relevant network license download web page:
  - For Workbenches released from January 2013 and later, (e.g. the Genomics Workbench version 6.0 or higher, and the Main Workbench, version 6.8 or higher), please go to: [https://secure.clcbio.com/LmxWSv3/GetLicenseFile](https://secure.clcbio.com/LmxWSv3/GetLicenseFile)
  - Paste in your license order ID and the host ID that you noted down in the relevant boxes on the webpage.
  - Click ‘download license’ and save the resulting .lic file.
  - Open the Workbench on your non-networked machine. In the Workbench license manager choose ‘Import a license from a file’. In the resulting dialog click ‘choose license file’ to browse the location of the .lic file you have just downloaded.
    - If the License Manager does not start up by default, you can start it up by going to the Help menu and choosing License Manager.
  - Click on the **Next** button and go through the remaining steps of the license manager wizard.

### 22.2 Server licenses

The method used to licensing a Server extension depends on the type of Server setup being used. There are three different server setups.

- **Single server setup** - A single machine is running the CLC Server software. Jobs are submitted to this server, which receives and executes them. In this setup, a single machine acts both as a master and an executor of jobs. Here, a single static license for the plugin is installed in the CLC Server software.
• **Job node setup** - More than one machine is running the CLC Server software. The system acting as the master server receives job requests and then submits these jobs to other machines, the job nodes, for execution. Here, a single static license is installed on each machine running the CLC Server software. That is, a static license is installed on the master node and on each job node.

• **Grid setup** - One machine runs the CLC Server software and receives job requests. It then submits these to a third party scheduler. The scheduler then chooses an appropriate grid machine, or node, to submit a given job to for execution. Here, a a single static license for the plugin is installed on the master server, and the same number of network plugin licenses as there are network gridworker licenses needs to be made available by installing these in the CLC License Server software.


### 22.2.1 Static license installation

In each of the server models described above, a static license is installed in the CLC Server on a master machine. In the case of a job node setup, static licenses are also installed on each machine acting as a job node.

Static licenses for the Server CLC Workbench are downloaded and installed into the licenses folder in the CLC Server installation area. Downloading a license is similar for all supported platforms, but varies in certain details. Please see the platform-specific instructions below for details on how to download a license file on the system you are running the CLC Server on. See section 22.2.5 for a description on how to download a license for a machine that does not have access to the internet.

For the master machine and for each machine in a job node setup:

1. Log on to the machine that is running the CLC Server.
2. Move into the CLC Server installation directory, where the license download script can be found.
3. Download and install the CLC Workbench license as described in the relevant section below.

### 22.2.2 Windows license download

License files are downloaded using the `licensedownload.bat` script. To run the script, right-click on the file and choose **Run as administrator**. This will present a window as shown in figure 22.16.

Paste the Order ID supplied by QIAGEN (right-click to Paste) and press Enter. Please contact ts-bioinformatics@qiagen.com if you have not received an Order ID.

Note that if you are upgrading an existing license file, this needs to be deleted from the licenses folder. When you run the `downloadlicense.command` script, it will create a new license file.
22.2.3 macOS license download

License files are downloaded using the `downloadlicense.command` script. To run the script, double-click on the file. This will present a window as shown in figure 22.17.

![Figure 22.17: Download a license based on the Order ID.](image)

Paste the Order ID supplied by QIAGEN and press Enter. Please contact ts-bioinformatics@qiagen.com if you have not received an Order ID.

Note that if you are upgrading an existing license file, this needs to be deleted from the `licenses` folder. When you run the `downloadlicense.command` script, it will create a new license file.

22.2.4 Linux license download

License files are downloaded using the `downloadlicense` script. Run the script and paste the Order ID supplied by QIAGEN Aarhus. Please contact ts-bioinformatics@qiagen.com if you have not received an Order ID.

Note that if you are upgrading an existing license file, this needs to be deleted from the `licenses` folder. When you run the `downloadlicense` script, it will create a new license file.

22.2.5 Download a static license on a non-networked machine

To download a static license for a machine that does not have direct access to the external network, you can follow the steps below after the Server software has been installed.

- Determine the host ID of the machine the server will be running on by running the same tool that would allow you to download a static license on a networked machine. The name of this tool depends on the system you are working on:
When you run the license download tool, the host ID for the machine you are working on will be printed to the terminal.

- Make a copy of this host ID such that you can use it on a machine that has internet access.
- Go to a computer with internet access, open a browser window and go to the relevant network license download web page:
  https://secure.clcbio.com/LmxWSv3/GetLicenseFile
- Paste in your license order ID and the host ID that you noted down earlier into the relevant boxes on the webpage.
- Click on 'download license' and save the resulting .lic file.
- Take this file to the machine with the host ID that you used when downloading the license file. Place it in the folder called 'licenses' that can be found within the CLC Server installation directory.
- Restart the CLC Server software.

22.2.6 Network license installation

Network licenses are necessary to run CLC Microbial Genomics Module analysis tasks on grid nodes. Network licenses are made available using a separate piece of software called the CLC License Server. This software is normally run as a service. CLC client software, such as Workbenches and gridworkers, contact the CLC License Server to obtain a network license when needed. For a description of how to download and install a license on a CLC License Server, please refer to the following section in the CLC License Server manual: http://resources.qiagenbioinformatics.com/manuals/clclicenseserver/current/index.php?manual=License_download.html

The same number of network plugin licenses as there are CLC gridworker licenses for the CLC Server setup are required. A license order ID is used when downloading a single license file. This license file includes information about how many network licenses are associated with the license order ID.


Index

Alpha diversity, 58
Alpha diversity measures, 60

Beta diversity, 61
Beta diversity measures, 62
Bibliography, 183
Borrow Workbench network license, 176
Combine schemes, 160
Contact information, 11

Download Amplicon-Based Reference Database, 155
Filtering, remove OTUs with low abundance, 32
Filtering, remove samples with low coverage, 22
Filtering, Result Metadata Table, 81
Fixed length trimming, 169
Functional annotation tools, 48

Hierarchical clustering of samples, 66
ID, license, 172
Importing OTU abundance tables, 31
Installation, 13

License, 14
ID, 172
non-networked machine, 179, 181
License server, 175
License server: use Workbench license offline, 176

Merging schemes, 160
Metadata, import, 69
MLST schemes, 136

Network license, 175
Network license: use Workbench offline, 176

OTU clustering, 23
Overview, 10

Pemanova, 63
Rarefaction analysis, 58
References, 183
Role, 77

Schemes
merge, 160
Server license, 16
Set Up Amplicon-Based Reference Database, 155
Stacked Bar Area, 30, 40, 55
Stacked Bar Chart, 30, 40, 55
Table, Metadata, 75
Table, MLST allele, 138
Table, MLST profile, 137
Table, Result Metadata, 78
Tree, k-mer, 132
Tree, SNP, 124

Uninstallation, 15
Use Genome as Result, 84

Venn diagram, 64
Visualization, abundance table, 28
Visualization, MLST schemes, 137
Vizualization, K-mer tree, 133
Vizualization, SNP tree, 130

Workflow, Data QC and OTU Clustering, 34
Workflow, Estimate Alpha and Beta Diversities, 35
Workflow, Map to specified reference, 86
Workflow, Type a Known Species, 105
Workflow, Type Multiple Species, 95

Zoomable Sunbursts, 30, 42, 56

184