



Deployment Manual

CLC WORKBENCHES

Manual for
CLC Workbenches: deployment and technical information
Windows, Mac OS X and Linux

June 15, 2017

This software is for research purposes only.

QIAGEN Aarhus
Silkeborgvej 2
Prismet
DK-8000 Aarhus C
Denmark



Contents

1	Introduction	5
1.1	Deployment strategies	5
1.2	Java	6
1.3	System requirements	6
2	Installation of Workbenches	9
2.1	Available installers	9
2.2	What does a Workbench installer do when it is run?	9
2.3	Installation modes - console and silent	10
3	License	11
3.1	Connecting to a CLC License Server from Workbenches	11
4	Plugins and resources	14
5	Workflows	15
6	Connecting to a CLC Server	16
7	Security policies	17
8	Storing and backing up data	19
8.1	Storing data	19
8.1.1	Data structure	19
8.1.2	Changing the default location	20
8.2	Back-up of data	21
8.3	Special configurations for large amounts of data	21
8.3.1	Temporary data	21

8.3.2 Disk space requirements	21
9 System resources	23
9.1 Setting the amount of memory available for the JVM	23
9.2 Setting the number of cores to use	24
10 Overview - where do we put things?	26
10.1 Computer-level information	26
10.2 Property files overview	26
10.3 User-level information	27
Index	27

Chapter 1

Introduction

The information in this manual is aimed at administrators who are responsible for Workbench installation, configuration and maintenance at a site. We include information here about installers, licenses and where the files associated with a Workbench installation are put.

For installing the *CLC Bioinformatics Database* and *CLC Server*, please refer to the installation chapter in the respective user manuals (see <http://www.qiagenbioinformatics.com/support/manuals/>).

At the moment, the CLC Workbenches are:

- *CLC Genomics Workbench*
- *Biomedical Genomics Workbench*
- *CLC Drug Discovery Workbench*
- *CLC Main Workbench*

In addition, there is the *CLC Sequence Viewer*, which is freely available.

In the first part of this manual, we take a closer look at deployment. Then the license system is discussed, followed by an explanation of the concept of plugins and workflows. Finally there is an overview chapter which tells you where to find all the different files in a client installation. This is useful when deploying the Workbench and also for back-up purposes.

Please contact AdvancedGenomicsSupport@qiagen.com if your questions about deployment are not answered in this manual.

1.1 Deployment strategies

Deployment strategies should be developed so that they fit your existing IT set-up.

The Workbench is self-contained and simple to configure. We provide here a brief summary of steps that can be taken to cut down on the work required to deploy Workbenches to a number of machines at a site. These steps can be tailored to suit the requirements of your own site. The steps below require you to log into each machine to install the Workbench, but cuts down on time spent configuring each machine separately with plugins, memory settings, default data

locations and temporary file locations. If you are working with a network license, this also saves time configuring these details on each machine. The rest of this manual is devoted to providing more details about Workbench installation, licensing and configuration.

1. On a single computer with the same OS as the target computers, complete a full installation of the Workbench including:
 - Set up the license server connection if you are using a network license setup. If you are using static licenses, you will need to download a license for this machine to complete the work described here and then activate licenses on each computer individually after the Workbench is installed on them.
 - Install the desired plugins. If you have any commercial plugins, set up the license server connection for these or license this particular machine copy if you are using static licenses. See the individual plugin manuals for instructions on how to do this.
 - Configure security policies, path where temporary data should be written to and the default data locations to be used to save data.
 - Specify proxy server information if needed for establishing connections to the external network.
2. Run the installer in silent mode (`-q`) on all the target computers.
3. Copy the following files from the installation in 1) to the target computers.
 - All files from the `settings` folder in the installation directory
 - `plugins` and `resource` folder from the installation directory

Creating a GHOST image or similar to copy to all the target computers is also a possibility, but there are a few routines performed by the installer that need to be taken into account, especially allocating memory (see section 9.1) and the creation of shortcuts and file associations.

1.2 Java

The Workbenches are Java software, which means there must be a Java Runtime Environment (JRE) to run the Workbench. The Workbench installers have a built-in JRE that is installed under the installation directory of the Workbench. The advantages of this include:

1. For computers that do not already have a JRE installed, no JRE needs to be downloaded and installed from another source.
2. For computers that already have a JRE installed, compatibility problems are avoided because the Workbench uses its own JRE.

The JRE used for running the CLC Workbench do not interfere with existing JREs on the computer.

1.3 System requirements

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

- Windows 7, Windows 8, Windows 10 or Windows Server 2012
- Mac OS X 10.9, 10.10, 10.11 and 10.12
- Linux: RHEL 6.0 and later, SUSE 13.1 and later
- 1 GB RAM required
- 2 GB RAM recommended
- 1024 x 768 display required
- 1600 x 1200 display recommended

The requirements for *CLC Genomics Workbench* and *Biomedical Genomics Workbench* are:

- Windows 7, Windows 8, Windows 10 or Windows Server 2012
 - Mac OS X 10.9, 10.10, 10.11 and 10.12
 - Linux: RHEL 6.0 and later, SUSE 13.1 and later. Minimum linux versions needed for the SRA download functionality: Red Hat 6, Fedora 12, and SUSE 11.
 - 64 bit operating system
 - 2 GB RAM required
 - 4 GB RAM recommended
 - 1024 x 768 display required
 - 1600 x 1200 display recommended
 - Intel or AMD CPU required
 - **Special requirements for the 3D Molecule Viewer**
 - **System requirements**
 - **3D Graphics Requirements**
 - * A graphics card capable of supporting OpenGL 2.0. Note that *CLC Main Workbench* only uses the GPU for the OpenGL 3D rendering. The GPU is not used to speed up molecular simulations.
 - * Updated graphics drivers. Please make sure the latest driver for the graphics card is installed .
 - **System Recommendations**
 - **3D Graphics 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 is recommended for working with large complexes.
- Indirect rendering (such as x11 forwarding through ssh), remote desktop connection/VNC, and running in virtual machines is not supported.

- **Special requirements for read mapping.** The numbers below give minimum and recommended memory for systems running mapping and analysis tasks. The requirements suggested are based on the genome size.
 - **E. coli K12 (4.6 megabases)**
 - * Minimum: 2 GB RAM
 - * Recommended: 4 GB RAM
 - **C. elegans (100 megabases) and Arabidopsis thaliana (120 megabases)**
 - * Minimum: 2 GB RAM
 - * Recommended: 4 GB RAM
 - **Zebrafish (1.5 gigabases)**
 - * Minimum: 2 GB RAM
 - * Recommended: 4 GB RAM
 - **Human (3.2 gigabases) and Mouse (2.7 gigabases)**
 - * Minimum: 6 GB RAM
 - * Recommended: 8 GB RAM
- **Special requirements for de novo assembly.** *De novo* assembly may need more memory than stated above - this depends both on the number of reads, error profile and the complexity and size of the genome. See http://resources.qiagenbioinformatics.com/white-papers/White_paper_on_de_novo_assembly_4.pdf for examples of the memory usage of various data sets.

Chapter 2

Installation of Workbenches

This chapter describes the Workbench installation process when using the standard installers.

For information about uninstalling a Workbench, please refer to our Frequently Asked Questions area "How can I uninstall a CLC Workbench?": <https://secure.clcbio.com/helpspot/index.php?pg=kb.page&id=279>

2.1 Available installers

There are installers available for each platform (Windows, Mac OS X and Linux). How to get installer files is described online in our Frequently Asked Questions area "Where can I get installer files for the software?": <https://secure.clcbio.com/helpspot/index.php?pg=kb.page&id=156>.

2.2 What does a Workbench installer do when it is run?

The installer performs the following tasks:

1. **Extracts and copies files to the installation directory** The Workbench is installed into a directory and is self contained. The suggested folder names reflect the software name and major version line, for example, for the *CLC Genomics Workbench* version 9.0, the default installation location offered would be:

Windows C:\Program files\CLC Genomics Workbench 9

Mac OS X /Applications/CLC Genomics Workbench 9

Linux /opt/CLCGenomicsWorkbench9

Central locations like these generally require administrator rights to install into them.

For minor updates, you will be asked whether you wish to update the existing installation (i.e. install new files in the same directory as previously) or install to a different directory. If you install a minor update to an existing installation location, licensing information will **not** need updating. If you install to a new directory, you will need to update the license information and install any plugins you wish to have. The License Manager will appear the

first time you run the Workbench if you have installed to a new directory. Updating the license in this case generally involves choosing the Upgrade from an existing Workbench option for static licenses or the Configure License Server Connection option if using a network license.

For major updates, the suggested folder to install into will reflect the new major version number. We recommend that for major upgrades, you install into a new location. That is, we recommend against choosing to install into the same folder as an existing, older version of the Workbench. For major updates, you will need to update your license information and install any plugins you wish to have for that version.

2. **Setting the amount of memory available** The installer investigates the amount of RAM on the machine during installation and sets the amount of memory that the Workbench can use. Read more in section 9.1.
3. **Shortcuts** On Windows and Mac systems, an option is provided during installation to create a shortcut for starting the Workbench. On Linux systems, this option is seen but has no effect.

2.3 Installation modes - console and silent

Console mode and silent mode are available when launching Workbench installers on the command line.

Console mode can be particularly useful when installing on remote systems. On Linux, this mode is enabled by using the option `-c`. On Windows the option is `-console`.

Silent mode allows the installation to be run in a hands off manner. Default answers to all prompts will be used. Silent mode is activated using the `-q` parameter. On Windows, the `-console` option can be appended as *the second parameter*, to ensure output to the console.

Specify an installation directory in combination with silent mode to specify a non-default installation directory and still not be prompted for responses to questions. The installation directory is specified using the `-dir` option.

So, on a Windows system, running the installation in silent mode, with console output and specifying the directory to install to as "c:\bioinformatics\clc" would look like:

```
CLCMainWorkbench_7_7.exe -q -console -dir "c:\bioinformatics\clc"
```

On a Linux system, a similar command to install to a directory "/opt/clcgenomicsworkbench9" would look like:

```
CLCMainWorkbench_7_7.exe -c -q -dir /opt/clcgenomicsworkbench9
```

Note for Windows: Both the `-console` and the `-dir` options only work when the installer is run in silent mode.

The `-q` and the `-console` options work for the uninstall program as well.

Chapter 3

License

There are two kinds of licenses for the Workbenches:

Static license Also known as a fixed license. A license order ID has to be activated against our server for each computer the software will be run on. A license file specific to a given machine needs to be present on that machine for the Workbench software to be run on it. Static licenses cannot be used on systems with more than 64 cores and remote use of software with a static license is not supported.

Network license Also known as a floating license. License Server software is installed at the site and this software hosts the network licenses. All license administration takes place via the license server software. Workbench software, usually installed on other machines, will contact the license server on startup to see if any licenses are available. Network licenses are needed if software will be run remotely or on systems with more than 64 cores. Network licenses are usually the best option for sites where there are many Workbenches.

Each Workbench manual contains a section about licensing. Workbench manuals can be obtained from <http://www.qiagenbioinformatics.com/support/manuals/>.

The CLC License Server manual explains how network licenses are administered: <http://resources.qiagenbioinformatics.com/manuals/clclikenseserver/current/>

Commercial plugins, or modules, use the same licensing system as the Workbenches, so the concepts described above also apply to plugin licenses. Network licenses are handled exactly the same way that Workbench network licenses are. How to download module static licenses in a Workbench is described in our Frequently Asked Questions area: <https://secure.clcbio.com/helpspot/index.php?pg=kb.page&id=351>.

3.1 Connecting to a CLC License Server from Workbenches

Setting up a connection from an individual Workbench to a CLC License Server is described in the Workbench manuals, which can be downloaded from <http://www.qiagenbioinformatics.com/support/manuals/>. It involves choosing the option "Configure License Server Connection" from the Workbench License Manager.

License server configuration information can be provided to Workbenches directly without the need to run each Workbench and use the graphical License Manager. After configuring a

single Workbench with the correct license server information, this license server information will be stored in a file called `license.properties` in the `settings` folder in the Workbench installation directory. This `license.properties` file can then be copied to the same relative location on other machines that will run a Workbench. When a Workbench is then started, it will look in this file, and if `useserver=true` then it will try to connect to the license server specified. No license configuration dialogs will be shown to the user.

The `license.properties` contains the following lines, with the information after the `̄` sign based on what is entered via the Workbench License Manager interface.

```
serverip=  
serverport=6200  
disableborrow=false  
autodiscover=true  
useserver=true  
username=  
usecustomusername=false
```


You can download a sample `license.properties` file at <http://resources.qiagenbioinformatics.com/deployment/license.properties>.

Chapter 4

Plugins and resources

Plugins can be either provided by QIAGEN Aarhus (see <http://www.qiagenbioinformatics.com/plugins/>) or can be custom-made plugins specific to your organization. Commercial plugins on a Workbench are often named modules, and commercial plugins on a CLC Server are often named extensions. Commercial plugins/modules/extensions require a license to be used. See section 3 for information about licensed plugins, also referred to as modules.

All information below pertains to both free and commercial plugins.

On a given workbench, plugins can be downloaded and installed directly in the **Plugin Manager**, or they can be installed from a plugin (.cpa) file by using the Install from File button at the bottom of the **Plugin Manager** (figure 4.1). The **Plugin Manager** () is started using the Plugins button on the Workbench toolbar or via the Plugins option in the **Help** menu.

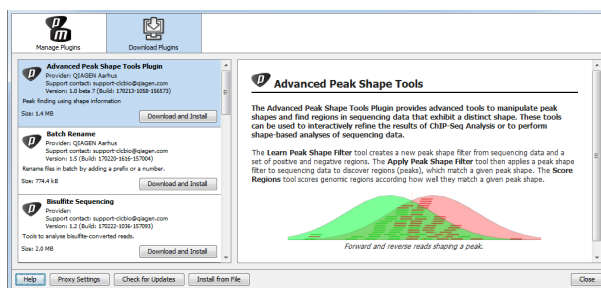


Figure 4.1: The Plugin Manager.

Plugins are installed to a folder called `plugins` in the installation area of a Workbench. When working with a large number of machines running a Workbench, one can avoid running the graphical Plugin Manager on each Workbench by:

- Configuring a single Workbench with the plugins desired using the Plugins Manager.
- Copying the `plugins` folder and its contents from under the Workbench installation area on that machine to the same relative location on other machines running the same type and version of Workbench software.

Chapter 5

Workflows

There is a graphical user interface to install workflows called **Workflows** (🖱️) which is invoked in the **Help** menu (see figure 5.1).

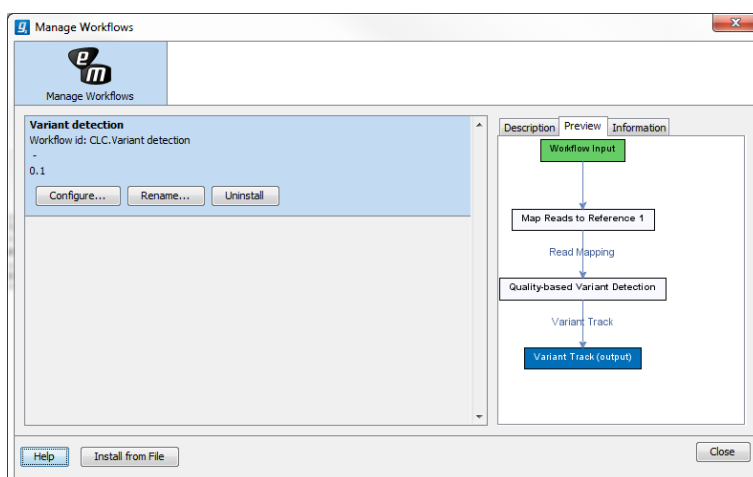


Figure 5.1: The workflow manager.

Workflows can be created in any workbench and distributed as an installer file that can be installed on other compatible Workbenches or CLC Servers.

A workflow installed on a Workbench is always installed on a per-user basis. The workflow definition is stored in the user home. See section 10.3.

The ability to install workflows can be disabled by the policy. See section 7.

When the workbench is part of a *CLC Genomics Server* setup, desired workflows can be installed on the CLC Server. All users logged into the server from their Workbench then have access to the centrally administered workflow. Updating an installed workflow then takes place just on the CLC Server, rather than on a per-user basis.

Chapter 6

Connecting to a CLC Server

Workbench users log into a CLC Server after installing the CLC Workbench Client Plugin into their Workbench and then going to the option File | CLC Server Login. The dialog that opens allows a user to enter the relevant details for the CLC Server to connect to.

These details can be pre-loaded into a Workbench by providing a file called `serverinfo.properties` in the `settings` folder in the Workbench installation directory.

If this file does not exist, the information that the user enters will be saved in their user settings. The username and password used is stored with the user settings.

A `serverinfo.properties` file contains the following:

```
port=7777  
host=hostname
```

This file can be copied to the same relative location on all machines running Workbenches that should be able to connect to the specified CLC Server.

You can download a sample `serverinfo.properties` file at <http://resources.qiagenbioinformatics.com/deployment/serverinfo.properties>.

Chapter 7

Security policies

The Workbench has a security policy configuration that enables administrators to restrict users' access to:

- **Tools accessing services on the internet.** This includes NCBI BLAST, NCBI and Uniprot Searches.
- **Notifications about updates.** Update notifications on new Workbench and plugin versions
- **Plugin management.** Installation of plugins.

The configuration is specified in a simple properties file called `policy.properties` that resides in the `settings` folder of the installation directory (e.g.

`C:\Program Files\CLC Main Workbench 6\settings` on Windows). Note that users without administrator access will not be able to change the contents of this file. Each of the following keys can be followed by either *allow* or *deny*:

workbench_version_check Controls whether notifications for Workbench updates should be shown.

plugin_version_check Controls whether notifications for plugin updates should be shown. Note that if `plugin_download` is not allowed, plugin update notifications will not be shown, regardless of this setting.

online_search Controls whether the **Download Genome** (🌐), **Search for Sequences at NCBI** (🌐), **Search for PDB Structures at NCBI** (🌐), **Search for Reads in SRA** (🌐) and **Search for Sequences in UniProt** (🌐) in the **Download** menu should be available (these can be used to search for and download sequences, reference data resources and 3D structures from NCBI and Swissprot/Uniprot).

online_ncbi_blast Controls whether all the tools performing BLAST at NCBI's servers should be available. This is **NCBI BLAST** (🌐) both from the **Toolbox** and from sequence selections.

ncbi_blast_download Controls whether the tool performing download of BLAST databases from NCBI's servers should be available. This is **Download BLAST databases** (🌐) tool from the **Toolbox**.

plugin_manage Controls whether the **Plugins manager** should be available for the user. Note that users can still install plugin updates if `plugin_download` and `plugin_version_check` are allowed.

plugin_file_install Controls whether this user should be allowed to install plugins and resources from a local file.

plugin_download Controls whether this user should be allowed to install CLC plugins and resources downloaded directly within the plugin manager dialog. This also includes manually checking for updated plugins in the **Plugins manager** dialog and also the automatic check for plugin updates at start-up.

workflow_manage Controls whether the user should be allowed to manage workflows.

workflow_file_install Controls whether the user should be allowed to install workflows from a file.

workflow_download Controls whether the user should be allowed to download and install workflows from the CLC workflows repository.

sequence_to_structure It is possible to remove the 'Link Variants To Structure' action by adding a line called: `sequence_to_structure = deny` to the `policy.properties`. This will disable the action, and prevent users from clicking 'Link to Structure' links in variant tables. It is also possible to remove the 'Download Find Structure Database' (DDWB) / 'Download 3D Protein Structure Database' (GWB + CRWB) action by adding the line called: `sequence_to_structure = deny` to the `policy.properties`.

usage_information_collection Controls whether anonymous information about the usage of the workbench is shared with QIAGEN in order to improve products. It is also possible to opt out of the usage information in Preferences in the workbench.

Per default, there is no `policy.properties` file, so everything is allowed. A commented sample file that you can download and edit is located at <http://resources.qiagenbioinformatics.com/deployment/3/policy.properties>. Download the file and place it in the `settings` folder, update the relevant values, and the new policy will take effect next time the Workbench is started.

Chapter 8

Storing and backing up data

This chapter explains how data is stored, gives general guidance on size of data, and outlines configurations needed for running analyses on large amounts of data.

8.1 Storing data

8.1.1 Data structure

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

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

This is explained visually in figure 8.1. The full path to the system folder can be located by mousing over the data location as shown in figure 8.2.

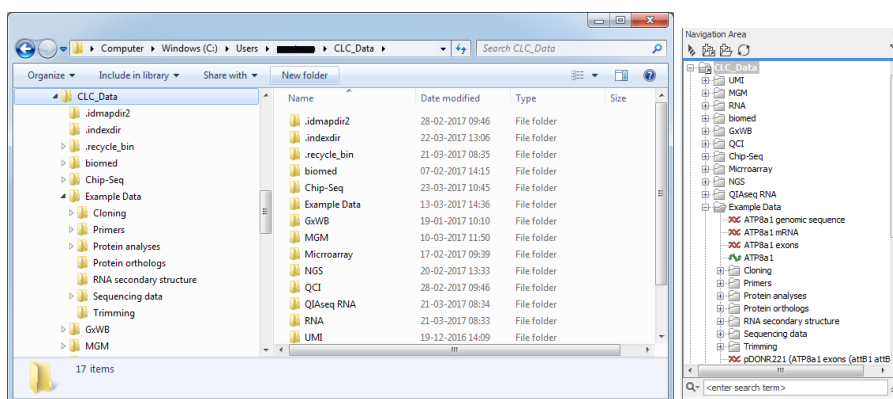


Figure 8.1: In this example the location called "CLC_Data" points to the folder at C:\Documents and settings\clcuser\CLC_Data.

If the Workbench is connected to a CLC Server, the server's locations will automatically show up when the user is logged in. This chapter does not deal with server locations - please refer to the server user manual at <http://www.qiagenbioinformatics.com/support/manuals/>.

The list of locations is stored in a file called `model_settings_300.xml` in the `settings`

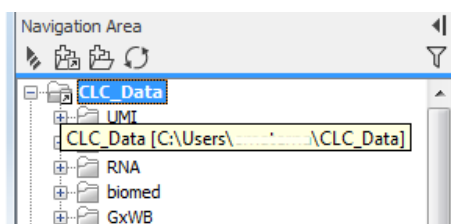


Figure 8.2: Mousing over the location called 'CLC_Data' shows the full path to the system folder, which in this case is `C:\Users\boester\CLC_Data`.

folder in the user home (see section 10.3). We do not recommend manual editing of this file, although it is standard xml.

8.1.2 Changing the default location

In some set-ups, storing data in the default location `CLC_Data` in the user home is not desired. This could be for roaming user profiles or in situations where there is a quota on this disk.

The default location that is used the first time the Workbench starts can be configured in the `path.properties` file that resides in the `settings` folder of the installation directory (e.g. `C:\Program Files\CLC Main Workbench 5\settings` on Windows). Add a line like this to the file to change the default location: The file should include one line like this:

```
datadir = c:\clcdata
```

The following variables can be used to construct the path to the desired location:

\$user the user name of the current user

\$home the home directory of the current user

\$product the short name of the workbench (example: `clcgenomicswb` or `clcmainwb`)

A few examples:

```
datadir = $home/CLC_Data          (default)
datadir = X:\clcstorage\%user     (separate disk / network mount in Windows)
```

Note that the folder does not need to exist - it will be created if needed. You can use both slash (normally used on Mac and Linux) and backslash (normally used on Windows) in the configuration file - the Workbench will use the appropriate one depending on the platform.

Note that the default location is only considered the very first time the Workbench starts. When the Workbench closes the first time, the `model_settings_300.xml` file is created and this is where it will look for the locations further on. Deleting this file will make the Workbench look for the `datadir` property in the `path.properties` file. The `model_settings_300.xml` file is located in the `settings` folder in the user home (see section 10.3).

You can download a sample `path.properties` file at <http://resources.qiagenbioinformatics.com/deployment/path.properties>.

8.2 Back-up of data

Since all data used in the Workbench is stored as files in the locations specified, a back-up procedure has to include all the locations. If the data needs to be restored from a back-up, simply copy the files back into the folder locations and start the Workbench.

Database locations needs a different back-up procedure.

Besides the data itself, user-level settings should also be included in the back-up (see section 10.3).

8.3 Special configurations for large amounts of data

Especially the *CLC Genomics Workbench* is often used with large amounts of data. This means that special configurations often need to be made. This concerns locations for temporary data and disk space in general.

8.3.1 Temporary data

The Workbench has a built-in cache system that intends to make sure that the Workbench does not run out of memory even for large data sets. During various processes such as assembly and RNA-Seq analysis, the Workbench often writes temporary files to the disk. Depending on the data set, these temporary files can take up a lot of disk space.

If there is not enough space in the default tmp directory, the tmp directory can be re-directed:

Create a text file called `path.properties` and save it in the `settings` folder in the Workbench installation directory. Please ensure this file does not have a ".txt" extension. The file should include one line like this:

```
tmpdir = /path-to-temp
```

Instead of "/path-to-temp" you write the absolute path to the new tmp directory. When the Workbench is restarted, it will then use the new directory for storing temporary data.

You can download a sample `path.properties` file at <http://resources.qiagenbioinformatics.com/deployment/path.properties>.

Note! It is imperative for acceptable performance that data transfer to the temp directory is not over a network connection. Since the Workbench will spend a lot of time writing and reading these files, disk speed has a great impact on overall performance when working with large data sets.

8.3.2 Disk space requirements

It is hard to give general guidance on disk space requirements, but we have made an example of a typical work flow for *CLC Genomics Workbench* to illustrate.

For calculating disk space for next-generation sequencing data you need to consider the following:

- Reads are imported and take up space as raw reads (see details below). Once imported, you can delete the original sequence file if you do not need it for other purposes.

- When the data has been assembled, either de-novo or against a reference, they take up space once again (this time more space since there is also information about where they map etc).
- Reference sequences also take up space.
- The computer doing the analysis needs space for tmp files. Once the assembly is done, the temporary files are deleted. The temporary files usually do not take up more space than the final result

The formulae giving the disk space usage:

Bytes per read: $28 + (\text{length of read name}) + 0.25 \times (\text{length of read})$

Note that you can discard read names during import.

If quality scores are present, add: $6 + (\text{length of read})$

If color space encoding is present, add: 7

As an example, a data set of 5.2 million 35 bp reads imported by *CLC Genomics Workbench* using the **Discard sequence names** option including quality scores takes up:

$$5,244,764 \times ((28 + 0 + 0.25 \times 35) + (6 + 35)) = 389 \text{ MB}$$

When mapped to a 4.7 Mbp annotated reference sequence, the mapping results takes up 473 MB.

Chapter 9

System resources

You can specify the number of CPU and the amount of memory that the java process of the Workbench is allowed to use.

9.1 Setting the amount of memory available for the JVM

When running the Workbench, the Java Virtual Machine (JVM) needs to know how much memory it can use. This depends on the amount of physical memory (RAM) and can thus be different from computer to computer. Therefore, the installer investigates the amount of RAM during installation and sets the amount of memory that the Workbench JVM can use to 50% of the computer's RAM or 50GB, whichever is the smaller. We recommend using these defaults.

The reason behind these defaults is explained below. First, we explain how to change these values, should you wish to.

Changing the memory setting for the Workbench java process You can raise or lower the amount of heap space that the Workbench java process is allowed to use by editing a setting in a file, as described below.

On **Windows** and **Linux**, this value is stored in a property file called `workbenchname.vmoptions` (e.g. `clcmainwb.vmoptions`). It is the `Xmx` setting. The content of this file looks like this:

```
-Xmx8000m
```

The value provided (8000m) is the amount of memory the Workbench java process is allowed to use. Here, 8000MB. The `vmoptions` file is placed directly under the installation folder. For example, on Windows, this would be at a location like:

```
C:\Program Files\CLC Main Workbench 7\clcmainwb.vmoptions.
```

On **Mac OS X**, the `-Xmx` value is stored in `vmoptions.txt` in the application bundle (Control-click the application and choose "Show Package Contents") from Biomedical Genomics Workbench 2.1, CLC Genomics Workbench 8.0, CLC main Workbench 7.6 and Drug discovery 2.0, and after. In older releases, the `-Xmx` value is stored at the same location in the package path but in a file called `Info.plist`. `Info.plist` file can still be used in the current version of the Workbench for customized `-Xmx` settings but the following needs to happen first:

1. add the following line into the `Info.plist` file, under the 'VMOptionArray' section:
`-Xmx4100m` (you can adjust "4100" to the memory setting you want)
2. the `-Xmx` setting inside the `vmoptions.txt` file needs to be removed, otherwise it will overwrite the setting in the `Info.plist`.'

If you do not wish to use the installer on each computer and plan to use an image instead, either all computers should have the same amount of RAM, or set the value used in the settings file to 50% of the computer with the smallest amount of RAM.

Default memory setting recommendations explained The suggested upper limit of 50GB is because there are limited gains in most circumstances if the value is increased beyond this java process beyond this and it ensures that the JVM does not spend too much time running garbage collection (GC) processes. The GC is a memory handling subsystem of the JVM, which scans through the currently used heap, and frees up memory by removing contents no longer in use. The larger the maximum heap, which is what this memory setting is affecting, the longer each check through the heap will take. The JVM has to suspend all other activities while running the GC. These suspensions are normally on the scale of milliseconds to a few seconds, and in this range, they are generally not noticeable. In combination with the GC subsystem of the JVM, the CLC Workbenches and Servers use a sophisticated caching system, moving unneeded data out of memory and into temporary disk storage. The result of this caching system means that there will not be any noticeable performance improvement with a Java heap space larger than 50GB, while GC pauses may become more noticeable.

The 50% of physical RAM suggestion is made because most CLC Workbenches and CLC Servers have some optimized external binaries responsible for compute-heavy tasks such as read mapping and de novo assembly. These external binaries are not affected by the maximum heap space limit. They request memory from the OS, just like any other executable. Thus, this suggested 50% limit for the heap space setting for the Java Virtual Machine (JVM) is to try to ensure that the machine will have sufficient free memory to run the external binaries.

9.2 Setting the number of cores to use

A number of the algorithms, particularly in Workbenches aimed at analyzing NGS data can use all the cores available on your system for large jobs to make the analysis as fast as possible.

You can restrict the number of cores that can be used to a predefined number via a properties file. To do this, create a text file called `cpu.properties` and save it in the `settings` folder in the Workbench installation directory. The file should include one line like this:

```
maxcores = 3
```

Here the maximum number of cores that the Workbench or binary algorithms (like read mapping or de novo analysis) would use would be 3.

After this file is in place and the Workbench is restarted, it will comply with this setting. Please note that this is not a guarantee that the Workbench will never use more than the number of cores specified, but that use of any more than this would be for very brief and infrequent peaks and

should not affect performance of other applications running on your system. The only exception is when a user starts several jobs to run concurrently on the Workbench.

You can download a sample `cpu.properties` file at <http://resources.qiagenbioinformatics.com/deployment/cpu.properties>.

Chapter 10

Overview - where do we put things?

This section provides an overview of the parts of a CLC Workbench installation. Some are at the level of the computer, i.e. shared by all users, whereas others are at a per-user level.

10.1 Computer-level information

In the *Workbench installation directory*, you will find the following:

Licenses The license information depends on what kind of license you use:

Static licenses Stored in the `licenses` folder (they are unique for each computer)

Network licenses Information about the license server connection is stored in the `license.properties` file in the `settings` folder.

Plugins are stored in the `plugins` folder and can be copied to other computers running the same version of the Workbench.

Memory allocation for the VM is stored in the `workbenchname(version_number).vmoptions` (e.g. `clcgenomicswb7.vmoptions`) file under the installation area of the Workbench on Windows and Linux. On Mac OS X, it is stored in `vmoptions.txt` in the application bundle (for example `/Applications/CLC Genomics Workbench 8/CLC Genomics Workbench 8.app/Contents/`).

Various property files are stored in the `settings` folder. See a full overview in section [10.2](#)

In addition, file associations for `.clc` files are stored in the registry database on Windows.

10.2 Property files overview

The following property files are all stored in the `settings` folder of the Workbench installation.

policy.properties for specifying if certain options should be removed from the Workbench (see section [7](#)).

path.properties for specifying default data location (see section [8.1.2](#)) and where to store temporary data (see section [8.3.1](#))

serverinfo.properties specify host and port for connecting to a CLC server (see section 6).

proxy.properties holds information about proxy server (when the Workbench needs access to online services). Read more here http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Network_configuration.html.

license.properties holds information about the host and port of a license server that should be used (see section 3.1).

cpu.properties specifies the maximum number of cores to be used by the Workbench (see section 9.2).

10.3 User-level information

The user-level information is found in the application data folder:

Windows Vista, 7 and 8 C:\Users\username\AppData\Roaming\CLC bio

Mac OS X User home/Library/Application Support/CLC bio

Linux \ \$HOME/.clcbio

In this folder, the following information may be useful to you:

User settings The user settings file stores information such as view settings, parameters, workspaces, user name and password to server log-ins and other settings customized by the user. The user settings file is found in the application data folder under settings/workbench name/version name. It is recommended to back up the user settings file. When upgrading to a new version of the Workbench, the user's old user settings file is copied by the Workbench the first time it is run.

Locations Information about which locations the user has added in the **Navigation Area** is stored in the model_settings_300.xml file in the settings folder.

Log Files If problems arise, these can be used by our support team to troubleshoot the issue. How to contact the Support team with the relevant log information is covered in our FAQ area at <https://secure.clcbio.com/helpspot/index.php?pg=kb.page&id=73>. If you wish to investigate the logs yourself, the key log files are called clc-workbench.log, output.log and error.log. Their location depends on the system you are working on. Details of the locations for each operating system can be found in our FAQ area at <https://secure.clcbio.com/helpspot/index.php?pg=kb.page&id=67>.

Workflows Workflows are saved in a workflows/workbench name/version name folder. When upgrading to a new version of the Workbench, the user's old workflows are copied by the Workbench the first time it is run, unless there are major changes to the workflow framework that break compatibility. If you are using a *CLC Genomics Server*, it can decrease overhead related to upgrading, and maintaining consistent versions of Workflows for all users, if Workflows are deployed and maintained on the Server rather than on the Workbench.

Index

- CLC Bioinformatics Database*, 5
- CLC Server*, 5
- 32-bit installer, 9
- 64 bit installer, 9
- Back up
 - user settings, 27
- Back-up, 21
- Block access to internet, 17
- Command-line installation, 10
- Cores, restrict usage, 24
- CPU, restrict usage of, 24
- Data storage, 19
- Data structure, 19
- Database
 - local, 19
- Floating license, 11
- GHOST image, 6
- Image, copy, 6
- Install
 - plugins, 14
 - workflows, 15
- Introduction, 5
- Java, 6
- JRE, Java Runtime Environment, 6
- JVM, Java Virtual Machine, 6
- License order ID, 11
- License server, 11
- Licenses, 11
 - for modules, 11
- Locations, 27
- Log files, 27
- Memory allocation, 23
- Module licenses, 11
- Network licenses
 - set-up on clients, 11
- Online tools, block access to, 17
- Plugin Manager, 14
- Plugins, 14
- Policy, 17
- Quiet installation, 10
- RAM, 23
- Security policies, 17
- Server connection, 16
- Silent installation, 10
- System requirements, 6
- System resources, 23
- Temporary data, 21
- User settings, 27
- VM, Virtual Machine, 6
 - .vmoptions, memory allocation, 23
- Workflows, 15
- Xmx argument, 23