



Deployment Manual

CLC WORKBENCHES

Manual for
CLC Workbenches: deployment and technical information
Windows, macOS and Linux

July 1, 2024

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	7
2	Installation of Workbenches	8
2.1	Getting Workbench installers	8
2.2	What a Workbench installer does when launched	8
2.3	Installation modes - console and silent	9
2.4	Uninstalling a Workbench	10
3	Licenses	11
3.1	Connecting to a CLC Network License Manager from Workbenches	11
4	Workbench plugins	13
5	Workflows	14
6	Connecting to a CLC Server	15
7	Security policies	17
8	Storing and backing up data	19
8.1	Default Workbench data storage	19
8.2	Temporary data	21
8.3	Backup of Workbench information	21
8.4	Disk space requirements	22

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 Per-computer Workbench information	26
10.2 Per-user Workbench information	27
10.3 Per CLC Data Location information	28
Index	29

Chapter 1

Introduction

This manual includes information about installers, licenses and where the files associated with a Workbench installation are put. It is intended for administrators responsible for Workbench installation, configuration and maintenance.

The currently available commercial CLC Workbenches are:

- *CLC Genomics Workbench*
- *CLC Main Workbench*

In this manual, we suggest methods for the efficient deployment of CLC Workbenches to multiple machines and describe the files and settings important to Workbench installations and their configuration.

Please contact ts-bioinformatics@qiagen.com if you have questions about Workbench deployment and setup that are not answered by the information in this manual.

For information about installing the *CLC Genomics Server*, please refer to the installation chapter in the CLC Server administration manual: https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Installing_running_Server.html.

1.1 Deployment strategies

The steps outlined here can be used when deploying identically configured Workbenches onto multiple machines, decreasing the time needed to configure each Workbench:

1. On a single computer that has the same OS as the other computers the Workbench will be installed on, complete a full Workbench installation.
2. Launch the newly installed Workbench and complete the initial Workbench setup by doing the following:
 - Set up the licensing. See section 3 for details.
 - Install any desired plugins. Plugins can be downloaded via the Workbench Plugin Manager if the Workbench can connect to the external network. Alternatively, plugin

files can be downloaded on a machine with external network access from <https://digitalinsights.qiagen.com/products-overview/plugins/>, and then copied to the machine with the Workbench, and installed using the "Install from File" functionality of the Workbench Plugin Manager.

- If you have any modules (commercial plugins) you plan to use for analysis work, the licensing can be set up for these now if you wish, but this step can also be done later¹.
 - For a network license setup, the relevant module licenses should be made available from your *CLC Network License Manager*.
 - For static licenses, a license file for the relevant products needs to be present on each machine where analyses will be carried out.

For further information about plugin installation and licensing, please refer to the individual plugin manuals, available from <https://digitalinsights.qiagen.com/products-overview/plugins/>.

3. Configure the settings to be applied to all Workbench users via properties files. See section 10.1 for a list of property files and links to the relevant chapters describing them.
4. Run the Workbench installer in silent mode (-q) on each of the other computers to install the Workbench on them. See section 2.3 for details.
5. Copy the following files from the initial Workbench installation to the corresponding location on the other computers.
 - All files under the `settings` folder of the installation directory.
 - The `plugins` folder and its contents from under the initial Workbench's installation area.

Another option for deployment is to create an image, which is then copied to all the target computers. With this option, though, there are a few routines performed by the Workbench installer that need to be taken into account, such as the creation of short cuts and file associations, and the setting of memory limits, which is described in section 9.1.

1.2 Java

The Workbenches are Java software, which means there must be a Java Runtime Environment (JRE) to run them. The Workbench installers have a built-in JRE that is installed under the installation directory of the Workbench. The advantages of this include that no JRE needs to be downloaded and installed from another source and that compatibility problems due to other JREs on a system are avoided because the Workbench uses its own JRE.

The JRE used for running a Workbench does not interfere with existing JREs on the computer.

¹For CLC Genomics Workbench 10.x, CLC Main Workbench 7.x, and Biomedical Genomics Workbench 4.x and as well as earlier versions of these products, you must install relevant module licenses at this point, even if you do not plan to use the initial installation for analysis work.

1.3 System requirements

The system requirements for Workbenches can be found within the individual Workbench manuals (<https://digitalinsights.qiagen.com/technical-support/manuals/>) or online at <https://digitalinsights.qiagen.com/technical-support/system-requirements/>.

Chapter 2

Installation of Workbenches

2.1 Getting Workbench installers

Workbench installers are available for Windows, macOS and Linux. How to get installer files is described in our Frequently Asked Questions area "Where can I get installer files for the software?": <https://qiagen.secure.force.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA41i000000L5uQCAS>.

2.2 What a Workbench installer does when launched

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

1. **Extract and copy files to the installation directory** The Workbench software is installed into a directory. It is self contained. The suggested folder name to install into reflects the software name and the major version line. For example, for a *CLC Genomics Workbench* with major version 24, the default installation location offered on each platform would be:

Windows C:\Program files\CLC Genomics Workbench 24

macOS /Applications/CLC Genomics Workbench 24

Linux /opt/CLCGenomicsWorkbench 24

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

General recommendations for installation locations

- **For minor updates**, you will be asked whether you wish to:
 - **Update the existing installation** *Generally recommended for minor updates.* New files will be installed into the same directory as the existing installation. Licensing information and installed plugins remain in place from the installation already present.
- OR

- **Install to a different directory.** Configuration will be needed after installation. E.g. licensing needs to be configured, any desired plugins will need to be installed, etc.
 - **For major updates.** The suggested installation directory will reflect the new major version number of the software. *Please do not install a new major version into the same folder as an existing, older version of the Workbench.* Configuration will be needed after installation. E.g. licensing needs to be configured, any desired plugins will need to be installed, etc.
2. **Set the amount of memory** The installer investigates the amount of RAM on the machine during installation and sets the amount of memory that the Workbench can use.
 3. **Establish shortcuts (optional)** On Windows and Mac systems, an option is provided during installation to create a shortcut for starting the Workbench. On Linux systems, this option is also presented, but it has no effect.

Read more about memory settings in section [9.1](#).

2.3 Installation modes - console and silent

Two installation modes are available to support efficient installation of the software.

- **Console mode** This mode is particularly useful when installing onto remote systems. On Linux, this mode is enabled by using the option `-c` when launching the installer from the command line. On Windows the option is `-console`.
- **Silent mode** This mode supports hands-off installation. Default answers to all prompts are used, although a non-default installation directory can be specified if desired (see below). Silent mode is activated using the `-q` parameter when launching the installer from the command line. On Windows, the `-console` option can be appended after `-q`, that is, as *the second parameter*, to ensure output to the console.

If desired, you can **specify the directory to install the software to** when running the installer in silent mode. Do this adding the `-dir` option to the command line.

On Windows, the `-console` and the `-dir` options only work when the installer is run in silent mode.

The following is an example of a command that would install the software into the directory "c:\bioinformatics\clc" on a Windows system using silent mode with console output. The same form of command works for any CLC installer.

```
CLCMainWorkbench_23_0_5_64.exe -console -q -dir "c:\bioinformatics\clc"
```

On a Linux system, a similar command to install into the directory "/opt/bioinformatics/clc" could look like:

```
./CLCGenomicsWorkbench_23_0_5_64.sh -q -c -dir /opt/bioinformatics/clc
```

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

2.4 Uninstalling a Workbench

For information about uninstalling a Workbench, please refer to our Frequently Asked Questions area "How can I uninstall a CLC Workbench?": <https://qiagen.secure.force.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA41i000000L5t8CAC>.

Chapter 3

Licenses

There are two kinds of licenses for the Workbenches:

Static licenses Static license files are placed under the "licenses" subfolder of the Workbench installation area. A static license is specific to the machine the Workbench software is installed on. 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 licenses Network licenses are hosted by a *CLC Network License Manager*. Workbenches are configured to contact the software hosting the licenses on startup to check if relevant licenses are available. Network licenses are a good option for sites where there are many Workbenches. In addition, to run Workbenches remotely or on systems with more than 64 cores, a network license is needed. Network licenses are sometimes also known as floating licenses.

If you are administering network licenses, please also refer to the *CLC Network License Manager* manual: <https://resources.qiagenbioinformatics.com/manuals/clcnetworklicensemanager/current/>.

Further information about configuring licensing is available in the individual Workbench manuals: <https://digitalinsights.qiagen.com/technical-support/manuals/>

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 at <https://qiagen.secure.force.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA41i000000L63tCAC>

3.1 Connecting to a CLC Network License Manager from Workbenches

When setting up multiple Workbenches, configuring the network license connection settings without launching each Workbench is an efficient option. To do this:

- Launch a CLC Workbench.

- In the License Manager of that Workbench, choose the option "Configure license manager connection" and provide the relevant information. Further details about this step are provided in the Workbench manuals: <https://digitalinsights.qiagen.com/technical-support/manuals/>
- Close the Workbench after the license connection has been configured.
- Find the file called `license.properties` under the `settings` folder in the Workbench installation area.
- Copy the `license.properties` into the `settings` folder of each of the other Workbench installations.

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

```
serverip=your.license.server
serverport=6200
disableborrow=false
autodiscover=true
useserver=true
username=
usecustomusername=false
```

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

Chapter 4

Workbench plugins

Plugins extend the functionality of CLC Workbenches and the CLC Genomics Server. Workbench plugins distributed by QIAGEN Aarhus can be downloaded via the Workbench Plugin Manager if the Workbench can connect to the external network. Alternatively, plugin files can be downloaded on a machine with external network access from <https://digitalinsights.qiagen.com/products-overview/plugins/>, and then copied to the machine with the Workbench and installed using the **Install from File** button at the bottom of the **Plugin Manager** (figure 4.1).

When a plugin is installed, it is put into a folder called `plugins` within the installation area of a Workbench. For a single Workbench, one would typically use the graphical Workbench Plugin Manager for installing plugins and modules. However, when administering a large number of Workbenches, it can be more efficient to:

- Install the desired plugins and modules on a single Workbench using the Workbench Plugin Manager.
- Copy the `plugins` folder and its contents from that Workbench's installation area to the same relative location on other machines running the same Workbench version.

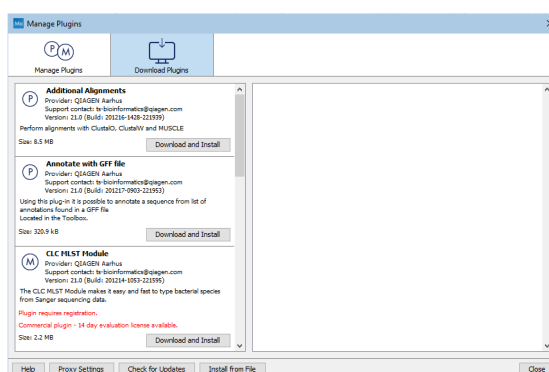


Figure 4.1: The Plugin Manager.

Tools delivered as part of a module, i.e. a commercial Workbench plugin, require a module license to be available. Please see section 3 for more on licensing.

Chapter 5

Workflows

Workflows can be installed and managed using the **Workflow Manager** in the Workbench. See figure 5.1.

The **Workflow Manager** can be invoked by choosing the option **Manage Workflows...** when you click on the **Workflows** button at the top right of the Workbench toolbar, or from under the **Utilities** menu.

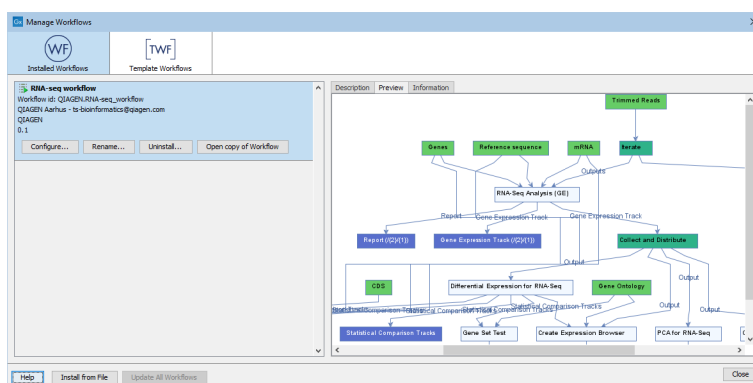


Figure 5.1: The workflow manager.

A workflow installed on a Workbench is always installed on a per-user basis. The workflow definition is stored in the user's home area. This is described further in section 10.2.

The ability to install workflows can be disabled by configuring a policy. See section 7 for further details about this.

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 would then take place on the CLC Server, rather than on a per-user basis.

Chapter 6

Connecting to a CLC Server

The CLC Server Connection dialog in CLC Workbenches can be used to log into or out of a *CLC Server*, as well as check on the status of a server connection. It is launched from the menu system by going to¹:

Connections | CLC Server Connection (S)

The location and port of CLC Servers that Workbenches are able to connect to can be pre-configured for all users of a Workbench on a given machine by placing a file called `serverinfo.properties` into the `settings` folder under the Workbench installation directory.

A `serverinfo.properties` file contains two lines for each *CLC Server* that should be listed in the drop down list within the CLC Server connection dialog in the Workbench.

For example, if 2 servers are available, the `serverinfo.properties` file would take the following form:

```
server.1.host = <hostname>
server.1.port = <portnumber>
server.2.host = <hostname>
server.2.port = <portnumber>
```

where `<hostname>` should be replaced with the name of the system the CLC Server is running on and `<portnumber>` by the port number of the CLC Server, typically 7777. The numbers in the parameter names (e.g. `server.1.x`, `server.2.x`) are used to specify the order the servers are presented in the dropdown list presented to users connecting from the Workbench. The order of the lines themselves does not matter. Up to 8 servers can be listed.

You can download an example `serverinfo.properties` file at

<https://resources.qiagenbioinformatics.com/deployment/serverinfo.properties>.

If a `serverinfo.properties` file is present under the `settings` folder of the Workbench installation, the name of the CLC Server entered for the `server.1.host` parameter in the file,

¹Prior to version 22.0, the CLC Server Connection option was under the File menu. This option was introduced in CLC Genomics Workbench 12.0 and CLC Main Workbench 8.1. Prior to that, the CLC Workbench Client Plugin needed to be installed to access this functionality.

and its port number, are initially presented in the dialog for all users of the Workbench on that system. When multiple servers are available, users can just select the one they want from the drop down list.

Machine level and user level CLC Server settings

To log into a *CLC Server*, Workbench users need to supply the *CLC Server* name and port, as well as a username and password. The server name, port and username will be remembered between Workbench sessions. This information is saved in their user settings file, `usersettings.xml`, whether or not a `serverinfo.properties` file exists.

Users can specify if they want their password to be stored. If they do, the encrypted password is saved in the `usersettings.xml` file. If they choose to save their password, they will also be able to specify if they want the Workbench to log into the *CLC Server* automatically each time the Workbench starts up.

If server information is present in both a user's `usersettings.xml` file and the central `serverinfo.properties` file, then the information from both is presented in the *CLC Server* Connection drop down list of servers. If the server information is identical in both files, only one item will be listed per server. If there is more than one server, the order they appear in the drop down list is determined by the information in the `serverinfo.properties` file.

Further details about the `serverinfo.properties` file

Only servers for which there is complete information (both a host and port setting) are presented in the *CLC Server* Connection drop down list of servers. If incomplete information is found in the `serverinfo.properties` file, (e.g. a host name setting with no corresponding port setting or vice versa), an error will be written to the *CLC Workbench* log file (`clc-workbench.log`), but any complete entries found in the property file will still be listed.

After a `serverinfo.properties` file is in place, server entries can be added or removed by directly editing the file.

Server details can also be added to `serverinfo.properties` indirectly: if the configured servers were read from the `serverinfo.properties` file **and** a user successfully logs into a *CLC Server*, **and** that user has write permission for the properties file, the server name and port used when logging in will be written back to the file as the first server in the list. I.e. that server's details will be entered as the `server.1.host`, and `server.1.port` settings, with other servers' details moved down in priority. This server will then be the first one listed in the drop down list in the *CLC Server* Connection dialog for all users of the Workbench on that system.

Chapter 7

Security policies

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






- **Tools accessing services on the internet**
- **Notifications about updates**
- **Plugin and Workflow management**


Policies are specified in a file called `policy.properties` that needs to be placed within the `settings` folder under the Workbench installation directory. Users without administrator access typically would not be able to change the contents of this file.

Each of the following keys within a `policy.properties` file can be followed by either = *allow* or = *deny*. No `policy.properties` file is present in a default Workbench installation, and thus by default, all the policies listed below are allowed.

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 set to "deny", plugin update notifications will not be shown, regardless of this setting.

online_search Controls access to online sequence and structure search engines and some databases from third party providers. Depending on the Workbench being used, this includes **Search for Sequences at NCBI** ()**, Search for PDB Structures at NCBI** ()**, Search for Sequences in UniProt** ()**, Download Genomes** () and **Search for Reads in SRA** ()**.**

online_ncbi_blast Controls whether tools performing BLAST at NCBI's servers should be available to users. This includes **NCBI BLAST** () run from the **Toolbox** and also from sequence selections.

ncbi_blast_download Controls whether the **Download BLAST databases** () tool, which allows download of BLAST databases from NCBI's servers, should be available.



plugin_manage Controls whether the **Plugin Manager** should be available. Users will still be able to install plugin updates if `plugin_download` and `plugin_version_check` are allowed.

plugin_file_install Controls whether plugins can be installed from a local file.

plugin_download Controls whether plugins can be downloaded and installed via the Download Plugins tab of the Workbench Plugin Manager. Setting this policy to deny also stops checks for updated plugins and the display of notification dialogs about available plugin updates during Workbench startup.

workflow_manage Controls whether the Manage Workflows dialog should be available.

workflow_file_install Controls whether workflows can be installed from a file.

sequence_to_structure Controls access to the **Link Variants To Structure** () and **Download 3D Protein Structure Database** () tools, as well as whether the 'Link to Structure' links in variant tables should be enabled or not.

usage_information_collection Controls whether anonymous information about the usage of the workbench is shared with QIAGEN, helping us improve our products. It is also possible to opt out of providing usage information on a per-user basis via a setting in the General section of the Preferences dialog.

welcome_center Controls whether the Workbench should contact the server that distributes content for the Welcome Center.

run_on_workbench_when_server_is_available Controls whether server-enabled tools, run directly or in the context of a workflow, can be run on a Workbench when it is connected to a CLC Server. It has no effect on tools that can be run only on a CLC Workbench or on the ability to download reference data to a CLC Server CLC_References location using the Reference Data Manager.

A sample `policy.properties` file can be downloaded from <https://resources.qiagenbioinformatics.com/deployment/policy.properties>. This can then be placed in the `settings` folder and edited to contain the relevant values. 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, outlines configuration changes useful for setups where analyses will be run on large amounts of data, and gives general guidance on backups and data sizes.

Where Workbench location information is stored

The list of Workbench CLC data locations is stored in a file called `model_settings_300.xml` located in the user level `settings` folder for the Workbench. This location depends on the operating system being used. Please see section 10.2 for details. We recommend that this file is **not** manually edited. See section 8.1 for information about changing default data locations for all users of a Workbench.

The `model_settings_300.xml` is created when the Workbench is shut down for the first time. Data location information is read from this file when starting up the Workbench again.

8.1 Default Workbench data storage

The default CLC location for storing data in Workbenches is called `CLC_Data`. An additional default location, `CLC_References`, is primarily intended to store reference data downloaded or imported using the CLC Genomics Workbench's Reference Data Manager. Folders of these names are created in the user's home area when the Workbench is started up if they do not already exist.

Changing the default data location for a single user from within a running Workbench is described at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Data_structure.html. Changing the location of the `CLC_References` folder for a single user from within a running CLC Genomics Workbench is described at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=References_management.html.

The remainder of this section describes making changes that apply to all users of a given Workbench installation.

Changing default data locations for a Workbench installation

The location of default Workbench CLC locations can be specified in a file called `path.properties`

placed in the `settings` folder in the Workbench installation area. If this file present before the Workbench is started up, the folder(s) to use as default locations are created in the locations specified in that file.

The relevant settings in the `path.properties` file are:

- **datadir** Used to specify the *full path* to the default data location to use (instead of `CLC_Data`).
- **refdir** Used to specify the full path to a folder called `CLC_References`. Any path provided that does not end in `CLC_References` will have a folder called `CLC_References` created under the folder specified. Relevant only for CLC Genomics Workbenches.

To completely avoid the creation of the original default locations `CLC_Data` and `CLC_References` on a system where CLC Workbenches have not previously been run:

- Install the Workbench but do not start it up.
- Place the `path.properties` file with the data location information in the `settings` folder in the Workbench installation area.
- Start the Workbench.

On start up, the Workbench will look for the data location settings in the `path.properties` file. If the default folder(s) specified in that file do not already exist, they will be created.

To update default data locations on systems where CLC Workbenches have been run previously:

- Ensure CLC Workbenches are not running on the system.
- Delete the `model_settings_300.xml` located in the user level `settings` folder for each Workbench user on the system.
- Place the `path.properties` file with the data location information in the `settings` folder in the Workbench installation area.

On start up, the Workbench will look for the data location settings in the `path.properties` file. If the default folder(s) specified in that file do not already exist, they will be created. Folders on the file system created by the Workbench earlier, e.g. `CLC_Data` and `CLC_References` folders in user home areas, will need to be manually deleted if they are no longer needed.

Note: The `model_settings_300.xml` file contains the locations of all CLC Workbench data locations. If you delete this file, then any non-default locations will need to be manually added to the Workbench again.

The following variables can be used when specifying the desired path:

\$user the user name of the current user

\$home the home directory of the current user

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

You can use either forward slashes (normally used on Mac and Linux) or backslashes (normally used on Windows) in the configuration file. The Workbench will use the appropriate type of slash character depending on the platform.

Examples:

```
datadir = $home/CLC_Data
datadir = X:\clcstorage\%user
refdir = X:\clcstorage\%user\CLC_References
refdir = /scratch/%user
```

In the above examples, the first is the standard default for a CLC Workbench. The last is equivalent to specifying `/scratch/%user/CLC_References`.

A sample `path.properties` file is available from

<https://resources.qiagenbioinformatics.com/deployment/path.properties>.

8.2 Temporary data

CLC Workbenches write temporary files to disk during many analyses, and by default, they are written to the standard location for temporary files on the machine. Depending on the data set, these files can take up a lot of disk space.

If space in the default temporary files directory is limited, a different location can be specified.

The temporary files directory must be on an internal drive on the system the Workbench is installed on.

To use a non-default location for temporary files generated by a CLC Workbench, 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 name with an additional extension, e.g. ".txt".

In the `path.properties` file, add a line like:

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

replacing `</path-to-temp>` with the absolute path to the new location.

Restart the Workbench for the new setting to take effect.

A sample `path.properties` file is available from

<https://resources.qiagenbioinformatics.com/deployment/path.properties>.

Further details: Performance is heavily dependent on constant, fast access to temporary files. Locations accessed over a network connection and locations on an external drive are thus not appropriate for this purpose.

8.3 Backup of Workbench information

We recommend backing up the following areas related to a Workbench installation:

Data: Workbench Data Locations are folders under which data are stored as files, often within a file hierarchy. If Workbench data needs to be restored from a backup, simply copy the restored files/folders back to the place they were originally located and start up the Workbench.

User level settings: See section 10.2 for a list of the user level settings files that can be useful to back up. When restoring from backup, close the Workbench if it is running and then place these files in the same place they were backed up from. The Workbench will use the restored settings when next started up.

Global settings: If customized policy files have been created, these may be useful to back up. Any that exist will be located under the `settings` folder under the Workbench installation area. See section 10.1 for detail of files that may be present. When restoring from backup, place restored properties files into the `settings` of the Workbench, overwriting any earlier files of the same name. Workbenches must be restarted to pick up the new policy details.

8.4 Disk space requirements

It is difficult to give general guidance on disk space requirements, but below, we describe some considerations relating to space requirements for sequencing reads and mappings, as these are typically the largest datasets generated when working with NGS data.

- After import, the reads will be in a Sequence List. Details relating to the size of Sequence Lists are below. The original sequence file (e.g. fasta, fastq, etc.) is not needed by the Workbench after import is complete.
- After import, reference sequences take up space. The original reference sequence files (e.g. fasta, embl, genbank), are not needed by the Workbench after import is complete.
- Reads mapped against a reference take up space within the mapping results. This is in addition to space taken up by the reads stored in a Sequence List.
- References also take up space within mapping results.
- Space is needed for temporary files created during analyses. Once an analysis is complete, the associated temporary files are deleted. Temporary files usually do not take up more space than the final result of the analysis. We highly recommend that temporary files are written to an area on the same machine as the Workbench is installed on. See section 8.2 for further details.

The formula for disk space usage relating to read sequences in a Sequence List is:

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 to save some space.

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

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

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

For a stand-alone read mapping that included a 4.7 Mbp annotated reference sequence, this would take up 473 MB.

Chapter 9

System resources

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

9.1 Setting the amount of memory available for the JVM

The Workbench Java Virtual Machine (JVM) needs to know how much memory it can use. The default value is 50% of the computer's RAM or 50GB, whichever is smaller. This is determined during Workbench installation. We recommend sticking with the default value for reasons described later in this chapter. However, it can be changed.

If you plan install Workbenches on multiple machines using an image, then either all the target computers should have the same amount of RAM, or the memory value specified should be set to 50% of the RAM on the computer with the smallest amount of memory.

Changing the memory setting for the Workbench JVM

The memory limit is configured in a property file by giving a numeric value and unit after the setting `-Xmx`. For example, this could look like:

```
-Xmx8000m
```

with 8000m indicating that the maximum amount of memory the Workbench java process is allowed to use is 8000Mb.

If a new memory limit is configured, it will be applied the next time the Workbench is started up.

The location of the relevant property file depends on the operating system being used:

Windows and Linux The property file is called `<workbenchname>.vmoptions`, so for a CLC Main Workbench installation, this would be `clcmainwb.vmoptions`. It is found under the installation area of the Workbench. For example, on Windows, this would be a location like: `C:\Program Files\CLC Main Workbench 24\clcmainwb.vmoptions`. Edit the `-Xmx` setting in this file.

macOS The property file is called `vmoptions.txt` and can be found in the application bundle. Edit the `-Xmx` setting in this file.

To find this file in the application bundle, open a Finder window and navigate to the Workbench application. Right click on it, or click on it while holding the Ctrl key down (Ctrl + click), and choose "Show Package Contents" from the menu that appears. The `vmoptions.txt` is within the Contents folder¹.

If you prefer this sort of information to be configured within the `Info.plist` file, also available under the Contents folder. This can be done using the following steps:

1. Add the following line into the `Info.plist` file, under the 'VMOptionArray' section: `-Xmx4100m`, adjusting the numeric value to the amount of memory you want.
2. Remove the `-Xmx` setting from within the `vmoptions.txt`. If this is not done, this "-Xmx" value will overwrite the setting in the `Info.plist` file.

Default memory setting recommendations explained

We suggest an upper limit of 50GB for the memory allocation because there are generally limited gains if the value is increased beyond this, and this limit helps ensure that the JVM does not spend too much time running garbage collection (GC) processes.

The GC is a memory handling subsystem of the JVM that 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 suggestion to limit memory use to a maximum of 50% of the physical RAM 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, the 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

Some Workbench algorithms can use all the cores available on your system for large jobs to make the analysis as fast as possible. This is particularly relevant for some NGS analysis tools.

You can restrict the number of cores that can be used by:

- Creating a text file called `cpu.properties` and saving it under the `settings` folder in the Workbench installation directory.

¹For releases before CLC Genomics Workbench 8.0, Biomedical Genomics Workbench 2.1 and CLC Main Workbench 7.6, the `-Xmx` setting for macOS was in the `Info.plist` file, which is also under the Contents folder.

- Editing that file, entering a single line like this:

```
maxcores = 3
```

where 3 should be replaced by the maximum number of cores to be used by a given instance tool, like read mapping or de novo assembly. If several such analyses are run concurrently on the Workbench, each of these would be able to use up to this maximum number of cores if needed.

- Restart the Workbench.

Please note that configuring the `maxcores` setting is **not** a guarantee that the Workbench will never use more than the number of cores specified. However, the use of any higher number of cores would be for very brief and infrequent peaks and should not affect performance of other applications running on your system.

You can download a sample `cpu.properties` file from

<https://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 global, affecting all Workbench users on that computer, while some are set at a per-user level.

10.1 Per-computer Workbench information

Files relating to the following areas are, unless otherwise specified, located under the *Workbench installation directory* and affect all Workbench users on a given system.

Licenses License information depends on the kind of license being used:

Static licenses These are stored as files within the `licenses` folder. A license file is specific for the computer it was downloaded for.

Network licenses Information about the connection to a *CLC Network License Manager* is stored in the `license.properties` file under the `settings` folder. See section 3.1 for further details. Example file for download:

<https://resources.qiagenbioinformatics.com/deployment/license.properties>

Plugins Installed plugins and modules are stored in the `plugins` folder. The `plugins` folder and its contents can be copied to other computers running the same version of a Workbench.

Memory allocation This is configured in the file `<workbenchname>.vmoptions` file under the installation area of the Workbench on Windows and Linux. On macOS, it is stored in `vmoptions.txt` file in the application bundle. See section section 9.1 for details.

Property files A Workbench installation can be fine tuned by placing one or more of the following properties files under the `settings` folder of the Workbench installation area.

policy.properties This can be used to restrict particular Workbench functionality such as installing plugins or using certain tools that download information from online sites. See section 7 for details. Example file for download:

<https://resources.qiagenbioinformatics.com/deployment/policy.properties>

path.properties This is used to configure a non-default location for data and for temporary data. See sections 8.1 and 8.2 for details. Example file for download:

<https://resources.qiagenbioinformatics.com/deployment/path.properties>

proxy.properties Proxy server settings are configured in this file. To get a copy of this file, configure the proxy settings using the Preferences dialog of a Workbench, as described here: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Network_configuration.html.

A `proxy.properties` file is then created under the `settings` folder of the Workbench installation area. These settings apply to all users of the Workbench on that system. To apply the same proxy settings to other Workbench installations, simply copy this file to the corresponding location on the other machines.

To connect directly to a *CLC Server*, bypassing the proxy settings when connecting, add the IP address (not the host name) of the *CLC Server* in the `proxyexclude` field.

cpu.properties The maximum number of cores to be used by the Workbench is configured here. See section 9.2 for details. Example file for download:

<https://resources.qiagenbioinformatics.com/deployment/cpu.properties>

serverinfo.properties The host and port to use when connecting to a *CLC Server* are configured here. See section 6 for details. Example file for download:

<https://resources.qiagenbioinformatics.com/deployment/serverinfo.properties>.

The format of the `serverinfo.properties` file changed with *CLC Genomics Workbench 12.0* and *CLC Main Workbench 8.1*. Please refer to [Connecting to a CLC Server](#) for a link to an example file relevant for older versions of the software.

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

10.2 Per-user Workbench information

Workbench settings that apply to each user, as well as log files, are found under the user's application data folder. Particularly useful files and areas are described below.

User settings

The `usersettings.xml` file stores information such as view settings, parameters, workspaces, username and password for *CLC Server* connections and other settings customized by the user. When an upgraded version of the Workbench is launched for the first time, the user settings file for the most recent older version on the system is copied and used for the new Workbench installation.

The `usersettings.xml` file is found under:

macOS

```
/Users/<username>/Library/Application Support/CLC bio/Workbench/ \
settings/<wbname>/<version>/
```

Linux

```
<userhome>/.clcbio/workbench/settings/<wbname>/<version>/
```

Windows 10

```
C:\Users\username\AppData\Roaming\CLC bio\Workbench\ \
Settings\wbname\version\
```

Data Locations

Information about data locations a user has added via the Workbench **Navigation Area** is stored in the `model_settings_300.xml` file, found under:

macOS

```
/Users/<username>/Library/Application Support/CLC bio/Workbench/settings/
```

Linux

```
<userhome>/.clcbio/workbench/settings/
```

Windows 10

```
C:\Users\username\AppData\Roaming\CLC bio\Workbench\Settings\
```

Log Files

If problems arise, log files are useful to our Support team for troubleshooting the issue. How to contact the QIAGEN Bioinformatics Support team with the relevant log information is covered in our FAQ area at <https://qiagen.secure.force.com/KnowledgeBase/KnowledgeNavigatorPage?id=kA41i000000L5r9CAC>.

If you wish to investigate the logs yourself, the key log files are called `clc-workbench.log`, `output.log` and `error.log`. These files can be found under:

macOS

```
/Users/<username>/Library/Application Support/CLC bio/Workbench/log/
```

Linux

```
<userhome>/.clcbio/workbench/log/
```

Windows 10

```
C:\Users\username\AppData\Roaming\CLC bio\Workbench\log\
```

Workflows

Workflows are saved under:

macOS

```
/Users/<username>/Library/Application Support/CLC bio/Workbench/ \
workflows/<wbname>/<version>/
```

Linux

```
<userhome>/.clcbio/workbench/workflows/<wbname>/<version>/
```

Windows 10

```
C:\Users\username\AppData\Roaming\CLC bio\Workbench\ \
workflows\wbname\version\
```

10.3 Per CLC Data Location information

Dot files are present in each folder in each CLC Data Location and each CLC Server File System Location, including at the top level of each location. These internal files, used for housekeeping, should **not** be manually edited, moved or deleted.

In each folder in a CLC Data Location:

- **.clcinfo** Contains the ID of the folder. In recycle bins, this file also contains information about where the data was moved from, when it was moved and by whom.
- **.fileinfocache** Contains information about files and folders within the folder, used to make browsing in a Navigation Area more efficient.

- **.orderlist2** Contains information about the order elements should be displayed in a Navigation Area.
- **.acl** Contains permissions-related information for the data in that folder. (CLC Server File System Locations only)

At the top level of a CLC Data Location and each CLC Server File System Location, there are additionally the following two dot files:

- **.idmapdir2** Contains a mapping of the ID to the path for this location.
- **.indexdir2** A Lucene index database. Lucene is used for the (data) search functionality.

Index

- 32-bit installer, [8](#)
- 64 bit installer, [8](#)
- Back up, [21](#)
 - user settings, [27](#)
- Block access to internet, [17](#)
- Command-line installation, [9](#)
- Cores, restrict usage, [24](#)
- CPU, restrict usage of, [24](#)
- Data
 - backups, [21](#)
 - storing, [19](#)
 - temporary files, [21](#)
- Data location default, changing, [19](#)
- Data location dot files, [28](#)
- Data storage, [19](#)
- Disk space, [22](#)
- Floating license, [11](#)
- Image, copy, [6](#)
- Install
 - plugins, [13](#)
 - workflows, [14](#)
- 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](#)
- Log files, [28](#)
- Memory allocation, [23](#)
- Module licenses, [11](#)
- Network license, [11](#)
- Network licenses
 - set-up on clients, [11](#)
- Online tools, block access to, [17](#)
- Plugin Manager, [13](#)
- Plugins, [13](#)
- Policy, [17](#)
- Quiet installation, [9](#)
- RAM, [23](#)
- Security policies, [17](#)
- Server connection, [15](#)
- Settings
 - user level, [27](#)
- Silent installation, [9](#)
- System requirements, [7](#)
- System resources, [23](#)
- Temporary data, [21](#)
- User settings, [27](#)
- VM, Virtual Machine, [6](#)
 - .vmoptions, memory allocation, [23](#)
- Workflows, [14](#)
- Xmx argument, [23](#)