CLC **Cloud** Module

USER MANUAL

User manual for

QIAGEN CLC Cloud Module 25.0

Windows, macOS and Linux

March 25, 2025

This software is for research purposes only.

QIAGEN Aarhus AS
Kalkværksvej 5, 11.
DK - 8000 Aarhus C
Denmark

Contents

Chapter 1

Introduction to the CLC Genomics Cloud

This manual describes the CLC Genomics Cloud solution, including establishing the necessary infrastructure on AWS, installation and use of QIAGEN CLC Cloud Module 25.0 and Cloud Server Plugin 25.0, and information about submitting CLC jobs to the cloud for execution.

1.1 Overview of CLC Genomics Cloud

With access to a *CLC Genomics Cloud* setup, analyses can be submitted from a *CLC Workbench* or a *CLC Server* to run on AWS EC2 instances, with results saved to AWS S3. Access to AWS S3 is built into CLC software, making it simple to upload data to and download data from the cloud. Data to be analyzed can be local or on AWS S3.

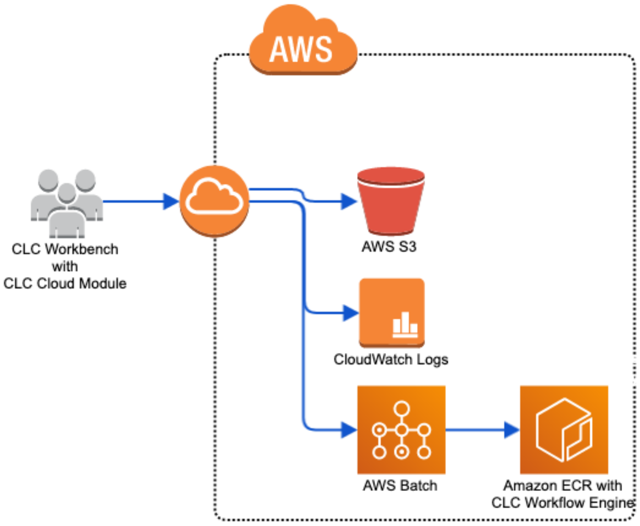


Figure 1.1: *The CLC Genomics Cloud is a combination of CLC software and AWS resources.*

With access to a *CLC Server* there is substantial potential for extending the functionality of CLC software using *External Applications*, whereby non-CLC tools can be launched using CLC software. In combination with a *CLC Genomics Cloud* setup, external applications can be launched directly from a *CLC Workbench*, via the Tools menu, or in a workflow context, to run on the cloud. See chapter 5 for details.

All traffic to and from AWS is encrypted using a minimum of TLS version 1.2. AWS credentials provided in a *CLC Workbench* are stored, obfuscated, within user configuration files. Those provided in a *CLC Server* are stored, obfuscated, in the server configuration files. All EC2 instances run in a security group that rejects incoming traffic that was not initiated from the EC2 host.

A *CLC Genomics Cloud* setup minimally consists of:

1. An AWS account with the relevant resources configured.
2. A *CLC Workbench* with an AWS Connection configured for access to that AWS account.
3. The QIAGEN CLC Cloud Module installed and licensed on a *CLC Workbench*.

With the above in place, analyses can be submitted from a *CLC Workbench* to run on the cloud (see chapter 3).

Jobs can also be submitted to run on the cloud via a *CLC Server* with the Cloud Server Plugin installed. Advantages of doing this are described in chapter 4.

The QIAGEN CLC Cloud Module is frequently updated. A detailed list of new features, improvements, bug fixes, and changes is available at `https://digitalinsights.qiagen.com/clc-cloud-module-latest-improvements/`.

CLC Genomics Cloud costs

The cost of a *CLC Genomics Cloud* setup consists of the costs of CLC software licenses and the costs of the AWS resources used (EC2 instances used for running analyses, AWS S3 used for data storage). There are no costs associated with setting up the *CLC Genomics Cloud* environment on AWS and there are no idle costs.

To submit analyses to run on AWS EC2 instances, you need the following CLC licenses:

- A license for a *CLC Workbench*. Jobs can also be submitted via a licensed *CLC Server*.
- A license for the CLC Cloud Module.

Pricing relevant to the AWS resources used by a *CLC Genomics Cloud* can be found at:

- **AWS EC2 On-demand instance pricing** `https://aws.amazon.com/ec2/pricing/on-demand/`
- **AWS S3 pricing** `https://aws.amazon.com/s3/pricing/`

You are billed directly by AWS for use of their resources.

1.2 CLC Genomics Cloud requirements

The requirements for a fully functional *CLC Genomics Cloud* setup are:

- Access to an AWS account where the relevant infrastructure has been made available. Specifically:
    - The *CLC Genomics Cloud* infrastructure, which is established using a CloudFormation template script, as described in chapter 2. Administrative privileges are required in the AWS account being used to set up this infrastructure.
    - At least one IAM user with appropriate permissions for submitting jobs to the *CLC Genomics Cloud*. The CloudFormation template script, described in chapter 2, includes the creation of one such user. Additional information about IAM users is also provided in that section.
    - At least one AWS S3 bucket for storing input data and results of analyses. These S3 buckets are not created by the scripts provided by QIAGEN. IAM users that will be submitting jobs to the *CLC Genomics Cloud* setup must have access to these buckets.
- A licensed *CLC Workbench*, with the CLC Cloud Module installed and licensed is required to submit jobs to the cloud for analysis.

  Installation of the CLC Cloud Module is described in section 8.

  Note: Files in AWS S3 buckets can be browsed and downloaded, and data can be uploaded to AWS S3, without the CLC Cloud Module. See section 3.4.
- If analyses will be submitted via a *CLC Server*, the Cloud Server Plugin must be installed and cloud presets configured.

  Working with the *CLC Server* is described in chapter 4.

General system requirements for CLC software are provided on https://digitalinsights.qiagen.com/technical-support/system-requirements/.

1.3 Contact information

QIAGEN CLC Cloud Module is developed by:

QIAGEN Aarhus A/S
Kalkværksvej 5, 11.
DK - 8000 Aarhus C
Denmark

https://digitalinsights.qiagen.com/

Email: ts-bioinformatics@qiagen.com

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

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

- Core product manuals https://digitalinsights.qiagen.com/technical-support/manuals/

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

1.4 Installing the CLC Cloud Module

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

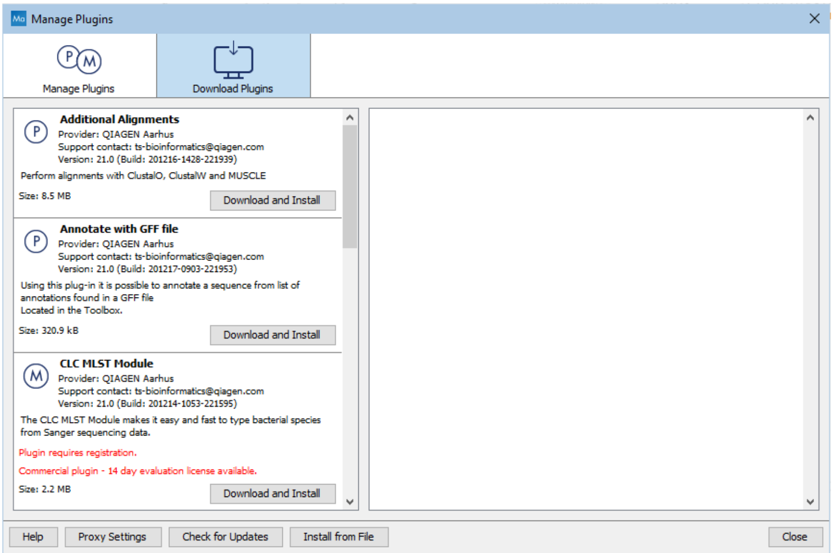
Plugins and modules are installed and uninstalled using the Workbench Plugin Manager. To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

Utilities | Manage Plugins... ( )

The Plugin Manager has two tabs at the top:

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

To install a plugin, click on the **Download Plugins** tab (figure 1.2). Select a plugin. Information about it will be shown in the right hand panel. Click on the **Download and Install** button to install the plugin.



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

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

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

Figure 1.2: *Plugins and modules available for installation are listed in the Plugin Manager under the Download Plugins tab.*

Accepting the license agreement

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

Installing a cpa file

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

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

Restart to complete the installation

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

1.4.1 Licensing modules

When you have installed the QIAGEN CLC Cloud Module and start a tool from that module for the first time, the License Assistant will open (figure 1.3).

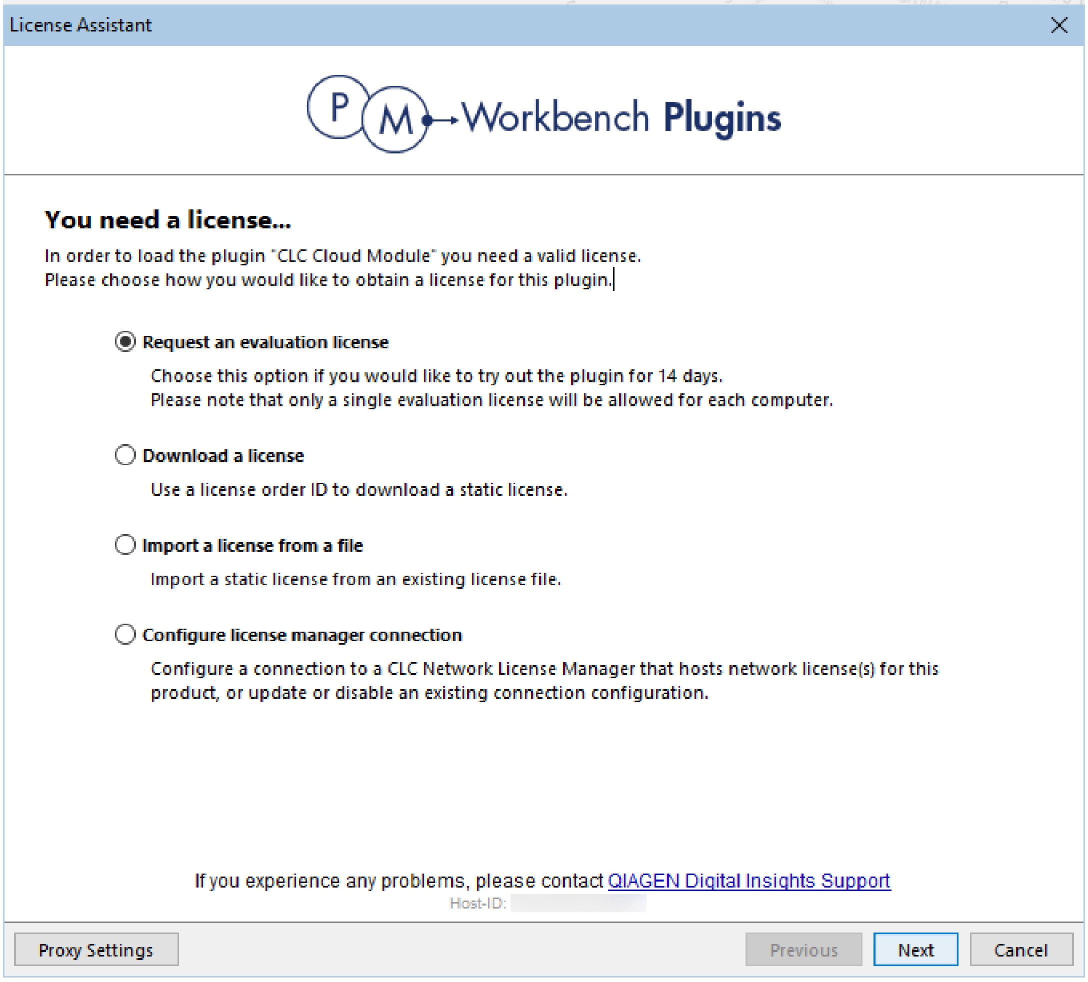
The License Assistant can also be launched by opening the Workbench Plugin Manager, selecting the installed module from under the Manage Plugins tab, and clicking on the button labeled *Import License*.

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

The following options are available:

- **Request an evaluation license.** Request a fully functional, time-limited license.
- **Download a license.** Use the license order ID received when you purchased the software to download and install a license file.
- **Import a license from a file.** Import an existing license file, for example a file downloaded from the web-based licensing system.
- **Configure license manager connection.** If your organization has a *CLC Network License Manager*, select this option to configure the connection to it.

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



You need a license...

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

● **Request an evaluation license**

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

○ **Download a license**

Use a license order ID to download a static license.

○ **Import a license from a file**

Import a static license from an existing license file.

○ **Configure license manager connection**

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

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

Host-ID: [redacted]

Figure 1.3: *The License Assistant provides options for licensing modules installed on the Workbench.*

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

1.4.2 Uninstalling modules

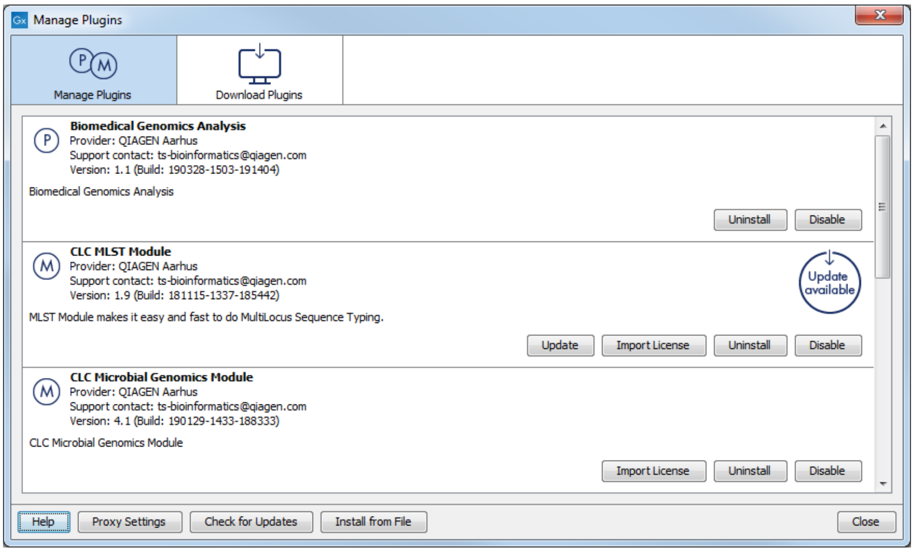
Plugins and modules are uninstalled using the Workbench Plugin Manager.To open the Plugin Manager, click on the **Plugins ( ) button** in the top Toolbar, or go to the menu option:

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

This will open the Plugin Manager (figure 1.4). Installed plugins and modules are shown under the Manage Plugins tab of the Plugins Manager.

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

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



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

Biomedical Genomics Analysis

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

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

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

CLC Microbial Genomics Module

Figure 1.4: *Installed plugins and modules are listed in the Plugins Manager under the Manage Plugins tab.*

Disabling a plugin without uninstalling it

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

1.5 Installing the Cloud Server Plugin

To submit jobs to run on the cloud via a *CLC Server*, the Cloud Server Plugin needs to be installed.

Download and install server plugins and server extensions

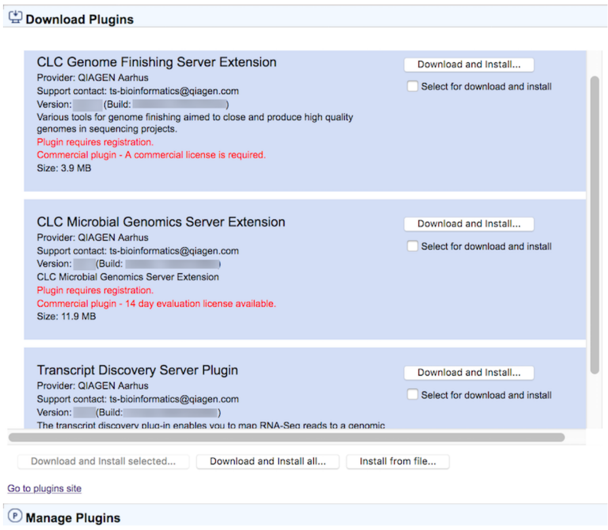
Plugins, including server extensions (commercial plugins), are installed by going to the **Extensions ( )** tab in the web administrative interface of the single server, or the master node of a job node or grid nod setup, and opening the **Download Plugins ( )** area (figure 1.5).

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

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

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

The *CLC Server* must be restarted to complete the installation or removal of plugins and server extensions. All jobs still in the queue at the time the server is shut down will be dropped and would need to be resubmitted. To minimize the impact on users, the server can be put into Maintenance Mode. In brief: running in Maintenance Mode allows current jobs to run, but no



Download Plugins

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

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

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

Figure 1.5: *Installing plugins and server extensions is done in the Download Plugins area under the Extensions tab.*

new jobs to be submitted, and users cannot log in. The *CLC Server* can then be restarted when desired. Each time you install or remove a plugin, you will be offered the opportunity to enter Maintenance Mode. You will also be offered the option to restart the *CLC Server*. If you choose not to restart when prompted, you can restart later using the option under the **Server maintenance (** ⚙ **)** tab.

Managing installed server plugins

Installed plugins can be updated or uninstalled, from under the **Manage Plugins** (Ⓟ) area (figure 1.6), under the **Extensions (** ✣ **)** tab.

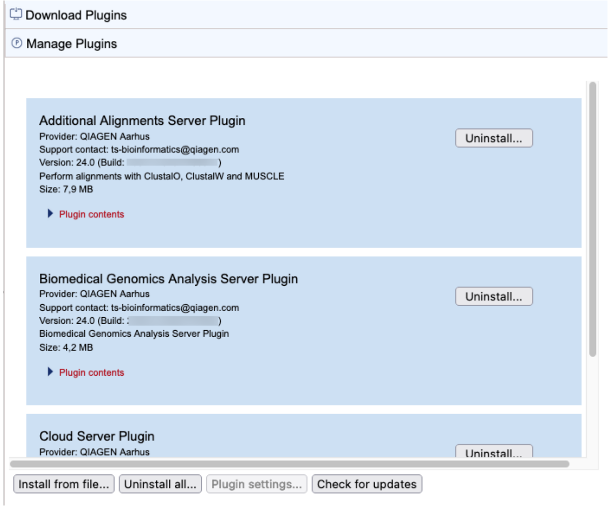
The list of tools delivered with a server plugin can be seen by clicking on the **Plugin contents** link to expand that section. Workflows delivered with a server plugin are not shown in this listing.

Plugin compatibility with the server software

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

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

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



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

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

Cloud Server Plugin
Provider: QIAGEN Aarhus

Figure 1.6: *Managing installed plugins and server extensions is done in the Manage Plugins area under the Extensions tab. Clicking on Plugin contents opens a list of the tools delivered by the plugin.*

Chapter 2

Setting up CLC Genomics Cloud resources on AWS

2.1 Setting up AWS resources

We recommend using the `cgc-standard.json` CloudFormation template, described below, to set up the standard resources needed for a *CLC Genomics Cloud* setup. Some configuration can be done when using this template. Additional AWS Batch queues can be added using another CloudFormation template afterwards. That template provides more configuration options (see section 2.2).

Below the description of the resources established using the `cgc-standard.json` CloudFormation template are detailed instructions of how to use the template, as well as information about the AWS IAM users created using the template.

Creating stacks using the AWS CloudFormation console is described in the AWS documentation at `https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/cfn-console-create-stack.html`.

Overview of standard CLC Genomics Cloud infrastructure on AWS

The `cgc-standard.json` CloudFormation template defines the resources needed for a *CLC Genomics Cloud*, which include:

- **Three AWS Batch queues** The queues are named `cgc-small`, `cgc-medium`, and `cgc-large`. When a user launches a job to run on the cloud from their *CLC Workbench*, they select a queue to send the job to from a drop down list of the available queues. Details of each of these queues are provided in section 8.1.
- **An S3 bucket for system files** The name of this bucket begins with "cgc-system-".

  This bucket is used by the CLC software for system files, including read mapper indexes. It is not intended for storing sample data or results. Details about the retention policies on this bucket are provided in section 8.3.

  The system bucket will not be accessible for browsing from a *CLC Workbench* and cannot be browsed by non-admin users logged into the *CLC Server* web client. It can be browsed by *CLC Server* admin users logged into the web client.

Note: S3 buckets for holding input data and results need to be created directly in AWS. They are not created by CloudFormation templates provided by QIAGEN.

- **Two AWS IAM users** One with properties supporting submission of analyses, and the other allowing only access to AWS S3 buckets. This is described in more detail below.

  When working with a *CLC Workbench*, access to the AWS Batch queues is determined by the access rights of the AWS IAM user configured in the AWS connection. With a *CLC Server*, access to AWS Batch queues can be fine tuned by setting group permissions on cloud presets, using the web administrative interface of the *CLC Server*, as described in section 4.3.

Creating CLC Genomics Cloud infrastructure on AWS

To set up the standard infrastructure on AWS for handling CLC jobs:

- Log into the AWS console as a user with privileges that allow the infrastructure described above to be created.
- Set the region to the one the AWS resources should be established in.
- Copy the URL below:

  `https://qiagen-clc-genomics-cloud-formation.s3.eu-central-1.amazonaws.com/cgc-standard.json`
- Go to the CloudFormation service and click on **Create stack**.
- In the **Create stack** step, keep "Choose an existing template" selected. In the "Specify template" step, keep the "Amazon S3 URL" option selected and paste the CloudFormation template URL you just copied into the "Amazon S3 URL" field.
- In the next step, specify a stack name and add a unique ID in the Parameters section.
- Review the *max vCPUs* setting for each queue.

  By default, the values allow for up to 10 EC2 instances to be launched. I.e. up to 10 jobs can be run in parallel. This value can be increased or decreased. We recommend it is not decreased below the number of cores designated for each job. These values can be found in the setting details provided in section 8.1.
- Review the *Disk size* setting for each queue.

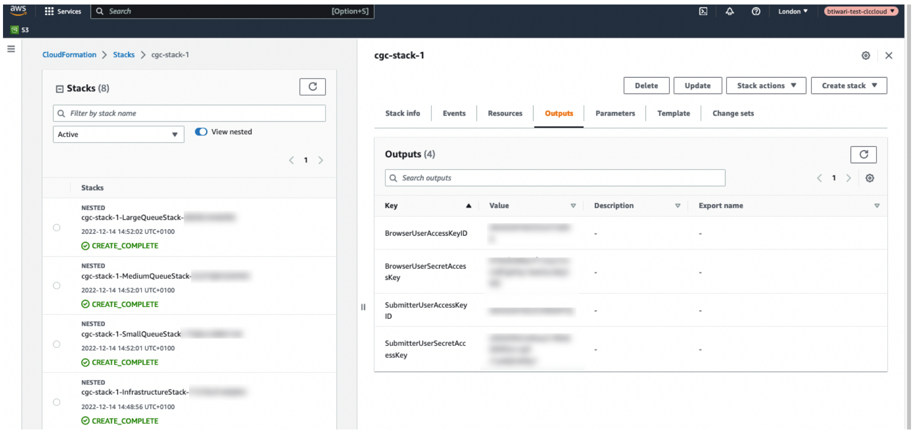
  This value refers to the space available for each job when it is running.
- Specify the *URL for custom plugin repository* if you have *custom* CLC plugins providing tools that should run on the cloud.

  See section 8.4 for the requirements for a custom plugin location.

  This field should be left blank if you require access only to plugins and modules distributed by QIAGEN. These are the plugins and modules listed in CLC Workbench and CLC Server Plugin Manager, or on the QIAGEN website: `https://digitalinsights.qiagen.com/products-overview/plugins/`
- Step through the rest of the stack creation. No other settings require configuration.
- When prompted, agree to the AWS conditions and click on the **Submit** button.

AWS IAM users

When stack creation is complete, go to the Outputs tab of the main stack to find the credentials for the AWS IAM users created (figure 2.1).



Stacks

NESTED
cgc-stack-1-LargeQueueStack-[redacted]
2022-12-14 14:52:02 UTC+0100
CREATE_COMPLETE

NESTED
cgc-stack-1-MediumQueueStack-[redacted]
2022-12-14 14:52:01 UTC+0100
CREATE_COMPLETE

NESTED
cgc-stack-1-SmallQueueStack[redacted]
2022-12-14 14:52:01 UTC+0100
CREATE_COMPLETE

NESTED
cgc-stack-1-InfrastructureStack-[redacted]
2022-12-14 14:48:56 UTC+0100
CREATE_COMPLETE

Key	Value	Description	Export name
BrowserUserAccessKeyID	[redacted]	-	-
BrowserUserSecretAccessKey	[redacted]	-	-
SubmitterUserAccessKeyID	[redacted]	-	-
SubmitterUserSecretAccessKey	[redacted]	-	-

Figure 2.1: *The credentials for the AWS IAM users created using the CloudFormation template are listed under the Outputs tab for the stack.*

AWS Connections using the "SubmitterUser" (`CgcSubmitterUser-<EnvironmentId>`) credentials allow CLC analyses to be submitted an AWS Batch queue for analysis. This user also has full access to AWS S3, read access to CloudWatch logs, and can list CloudFormation resources.

AWS Connections using the "BrowserUser" (`CgcBrowserUser-<EnvironmentId>`) credentials support listing S3 buckets and accessing bucket contents. Jobs cannot be submitted to run on AWS using these credentials.

AWS IAM user credentials are entered in AWS Connections in CLC Workbenches or the CLC Server, described in section 3.2 and section 4.2, respectively.

The full policy for each user can be viewed in the Identity and Access Management (IAM) area of the AWS Console.

AWS S3 buckets for storing input data and results

One or more AWS S3 buckets must be created for holding input data and results. These buckets must be created in the same AWS account and region that the AWS Batch queues were established in. Please refer to AWS documentation for details:

`https://docs.aws.amazon.com/AmazonS3/latest/userguide/create-bucket-overview.html`.

Note: The prefix `cgc-system-` should be considered reserved. Buckets given names starting with this term will not be visible in CLC Workbenches.

2.2 Adding more AWS Batch queues for CLC jobs

To add another AWS Batch queue for running CLC analyses using on-demand EC2 instances:

- Log into the AWS console as a user with privileges that will allow the infrastructure described above to be created.

- Set the region to the one the AWS resources should be established in.
- Copy the URL below:

  https://qiagen-clc-genomics-cloud-formation.s3.eu-central-1.amazonaws.com/cgc-queue-ondemand.json

- Go to CloudFormation and click on **Create stack**.
- In the **Create stack** step, keep "Choose an existing template" selected and in the "Specify template" step, keep the "Amazon S3 URL" option selected and paste the URL to the CloudFormation template you just copied into that field.
- Configure the settings for the new queue. This includes:
  - **vCPUs per job** We recommend setting this to the number of vCPUs available on the instance types you specify. If there are different numbers of vCPUs on the instance types specified, provide the lowest value.
  - **Total available memory per job** We recommend that this is set to the available amount of memory on the instance types you specify, minus 1000 MB. So for instance types with 32 GB of memory, the recommended value would be 31000.
  - **Amount of memory allocated for Java per job** We recommend that this is set to approximately half the total memory, up to a maximum of 50GB. So, with 31000 MB of total memory allocated, a reasonable value here would be 15500 MB. With 128000 MB total memory, a reasonable value would be 50000MB.
  - **max vCPUs** This value controls the number of EC2 instances that can be started up in parallel in this queue. For example, if the vCPUs per job were set to 16, then the value of 160 here would mean a maximum of 10 EC2 instances could be running at the same time in this queue.
  - **Allowed instance types** One or more EC2 instance types that can be started up when jobs are sent to this queue. Linux 64-bit (x86_64 / AMD64) instance types are supported. See the AWS documentation for details of the types available, https://aws.amazon.com/ec2/instance-types/ and their costs, https://aws.amazon.com/ec2/pricing/on-demand/.
  - **Disk size** Specify the size of the disk to allocate to each EC2 instance.
- When prompted, agree to the AWS conditions and click on the **Submit** button.

AWS Batch queues made using this template can be used by the AWS IAM "SubmitterUser" created using the standard template (see section 2.1).

When an AWS Batch queue is created using the CloudFormation templates provided by QIAGEN, it is mapped to an individual compute environment configured with the latest Amazon ECS-optimized AMI available at that time. Thus, the AMI version used when jobs are sent to queues established at different times may be different. See appendix section 8.2 for further details.

2.3 Adding more AWS IAM users

To create additional AWS IAM users with relevant permissions, choose the option to "Copy permissions from existing user" when creating the IAM user, and select one of the users created using the standard CloudFormation template as the user to copy.

Please refer to AWS documentation for further details:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users_create.html

2.4 Deleting or redeploying CLC Genomics Cloud infrastructure on AWS

Deleting an AWS Batch queue

To delete a particular AWS Batch queue, go to

> **AWS Batch | Job queues**

Select the queue you wish to delete and click on the Disable button.

This action may take a while. Once complete, select the queue again, and click on the Delete button.

Deleting CLC Genomics Cloud infrastructure

if you have run only the cgc-standard.json CloudFormation template to create your CGC Environment (and have not run other CloudFormation templates), then deleting the existing CGC infrastructure is usually a three step process:

1. **Empty the contents of the cgc-system bucket** Go to the S3 area of the AWS Console, select the cgc-system bucket and click on the **Empty** button. Non-empty system buckets cannot be deleted as part of the stack deletion.
2. **Delete the stack** Go to the CloudFormation area of the AWS Console, select the main stack, and click on the **Delete** button.
3. Delete old job definitions If any job was run using the older CGC resources, delete the old job definitions. Go to **AWS Batch | Job definitions** , select the job definition files with names that start with cwe and click on **Deregister**.

If stacks for additional job queues have been created, delete these stacks before carrying out the steps above. These additional stacks are linked as dependencies to the Infrastructure stack, which is a nested part of the CGC Environment.

Redeploying CLC Genomics Cloud Infrastructure

To redeploy CGC infrastructure on AWS:

1. **Delete the existing CGC resources** Please refer to the paragraph above.

2. **Deploy new CGC resources** Follow the instructions provided earlier in this chapter.

New user credentials will need to entered for AWS Connections in CLC Workbenches and CLC Servers. Cloud presets on CLC Servers will also need to be configured.

Chapter 3

Using a CLC Workbench to work on the cloud

This chapter describes configuring and using a *CLC Workbench* to submit jobs for execution on AWS, as well as working with data on AWS S3, including uploading data to AWS S3.

3.1 Prerequisites for working on the cloud using a Workbench

Submitting analyses to run on AWS

To submit jobs to run on AWS from a *CLC Workbench* you need:

- A licensed *CLC Workbench*.
- The CLC Cloud Module installed on the Workbench and licensed (see chapter 1.4).
- An AWS Connection configured with AWS IAM user credentials for a user with permission to submit analyses to run on AWS (see section 3.2).

An option called *CLC Genomics Cloud* is enabled in launch wizards when the setup is ready (see section 3.3)..

3.2 Configuring an AWS Connection in the Workbench

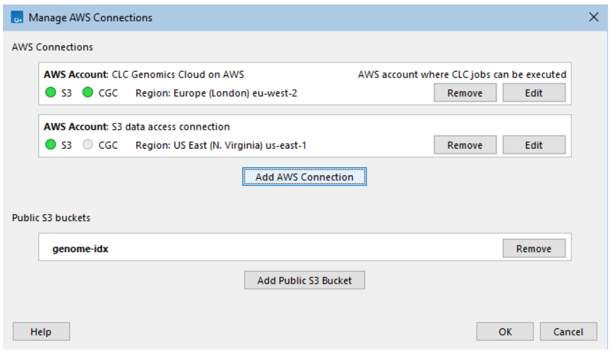
To configure an AWS Connection or to configure access to public AWS S3 buckets, go to:

> **Connections** | **AWS Connections** (☁)

Already configured AWS connections and their status, and public S3 buckets are listed (figure 3.1). Editing or removal of these configurations is done from here.

Configuring an AWS Connection

To configure a new AWS Connection, click on the **Add AWS Connection** button and enter the following information in the dialog (figure 3.2):



AWS Account: CLC Genomics Cloud on AWS — AWS account where CLC jobs can be executed

● S3 ● CGC Region: Europe (London) eu-west-2

AWS Account: S3 data access connection

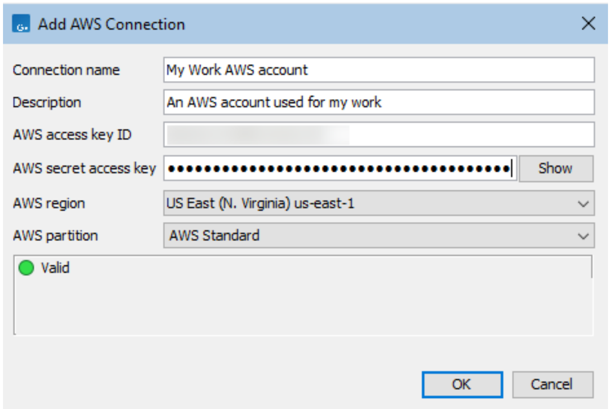
● S3 ○ CGC Region: US East (N. Virginia) us-east-1

genome-idx

Figure 3.1: *The configuration dialog for AWS connections. Here, two valid AWS connections, their status, and a public S3 bucket are listed.*

- **Connection name**: A short name of your choice, identifying the AWS account. This name will be shown as the name of the data location when importing data to or exporting data from Amazon S3.
- **Description**: A description of the AWS account (optional).
- **AWS access key ID**: The access key ID for programmatic access for your AWS IAM user.
- **AWS secret access key**: The secret access key for programmatic access for your AWS IAM user.
- **AWS region**: An AWS region. Select from the drop-down list.
- **AWS partition**: The AWS partition for your account.

The dialog continuously validates the settings entered. When they are valid, the Status box will contain the text "Valid" and a green icon will be shown. Click on **OK** to save the settings.



Connection name: My Work AWS account

Description: An AWS account used for my work

AWS access key ID:

AWS secret access key: ••

AWS region: US East (N. Virginia) us-east-1

AWS partition: AWS Standard

● Valid

Figure 3.2: *Configuration of an AWS Connection in a CLC Workbench*

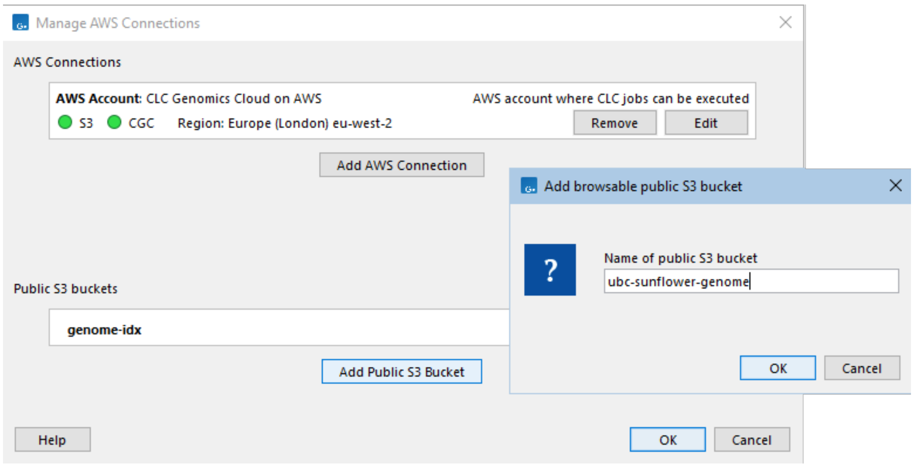
AWS connection status is indicated using colors. Green indicates the connection is valid and ready for use. Connections to a *CLC Genomics Cloud* are indicated in the CGC column (figure 3.1). To submit analyses to the *CLC Genomics Cloud*, the *CLC Cloud Module* must be installed and a license for that module must be available.

AWS credentials entered are stored, obfuscated, in Workbench user configuration files.

Note: Multiple AWS Connections using credentials for the same AWS account cannot be configured.

Adding a public S3 bucket

To add a public bucket, click on the **Add Public S3 button** and provide the public bucket name (figure 3.3).



AWS Account: CLC Genomics Cloud on AWS

Region: Europe (London) eu-west-2

genome-idx

Name of public S3 bucket: ubc-sunflower-genome

Figure 3.3: *Provide a public AWS S3 bucket name to enable access to data in that public bucket.*

3.3 Submitting analyses to AWS from a Workbench

Once the prerequisites described in section 3 are in place, the *CLC Genomics Cloud* option will be enabled in the launch wizards of CLC workflows and tools[1]. The available AWS Batch queues are listed in a drop-down menu (figure 3.4). After selecting a queue, mouse over its name to see information about the settings for that queue (figure 3.5).

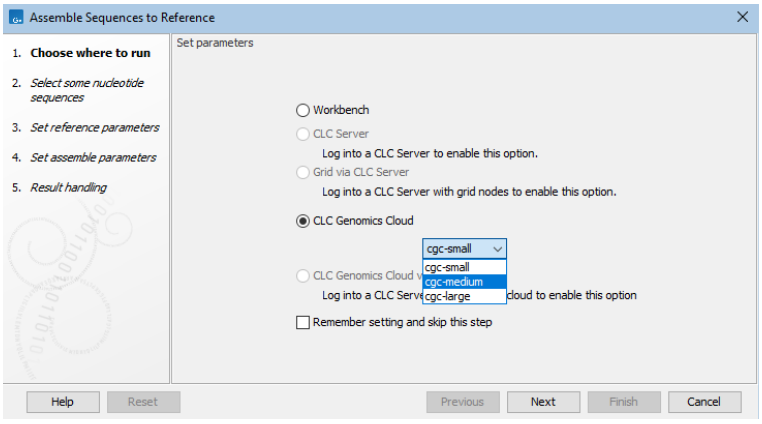
Select a folder in AWS S3 for results that has *not* previously been used to store results. When a non-empty folder is selected, a warning is shown. Files of the same name as files created by the analysis will be overwritten. This includes the `workflow-result.json` file, which contains a list of the results generated by a job and links to the `Execution Log` (described in chapter 7). If the `workflow-result.json` file is overwritten, functionality in CLC software for finding results for a given job will not work, even though the relevant files may be present and locatable on AWS S3, for example using the Remote Files tab.

Ensure that all processes for cloud jobs have proceeded beyond transferring job information and data before closing the Workbench. If closed earlier, the job will fail. When input data is already on S3, this phase is very quick as only information about the job is being sent to AWS. If the input data is not on AWS S3, the first activity after launching the analysis will be to transfer the data to AWS S3, so this phase can take some time. See section 6.1 for further details.

Software version used in the cloud When submitting jobs from a *CLC Workbench*, the version of

---

[1] Only tools that can be used within workflows can be submitted to run on a CLC Genomics Cloud. Tools that are not workflow-enabled cannot be run on the cloud.



○ Workbench

○ CLC Server

Log into a CLC Server to enable this option.

○ Grid via CLC Server

Log into a CLC Server with grid nodes to enable this option.

● CLC Genomics Cloud

cgc-small

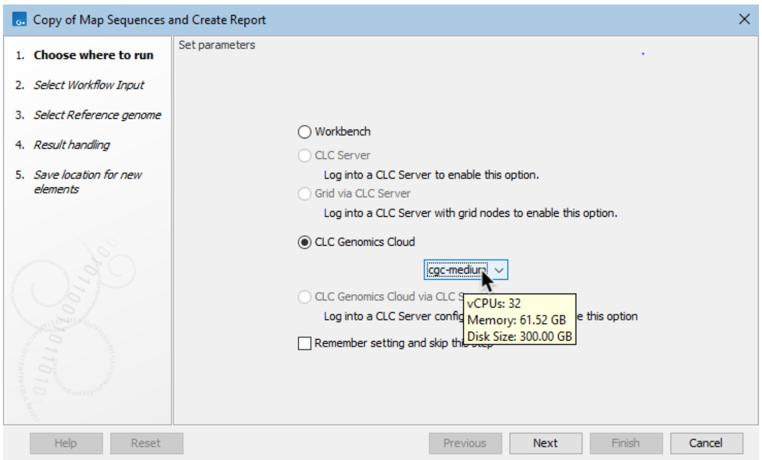
cgc-small
cgc-medium
cgc-large

○ CLC Genomics Cloud v

Log into a CLC Serve cloud to enable this option

☐ Remember setting and skip this step

Figure 3.4: *To submit jobs to the cloud, select the CLC Genomics Cloud option in the launch wizard and then select from the list of available AWS Batch queues in the drop-down menu.*



○ Workbench

○ CLC Server

Log into a CLC Server to enable this option.

○ Grid via CLC Server

Log into a CLC Server with grid nodes to enable this option.

● CLC Genomics Cloud

cgc-medium

○ CLC Genomics Cloud via CLC S

Log into a CLC Server config e this option

☐ Remember setting and skip th

vCPUs: 32
Memory: 61.52 GB
Disk Size: 300.00 GB

Figure 3.5: *When the mouse cursor hovers over the selected queue, details about the computational resources associated with that queue are shown in a tooltip.*

the CLC software used in the cloud is the same as the *CLC Workbench* version. When submitting jobs that include tools provided by a plugin, the plugin version used in the cloud job is the same as the version installed on the *CLC Workbench* used to submit the job. See section 8.2 for further details.

3.4 Working with AWS S3 using the Remote Files tab

Browsing data in AWS S3 using the Remote Files tab

Available AWS S3 buckets are listed under the **Remote Files** tab, to the right of the Navigation Area tab (figure 3.6).

If AWS S3 buckets are available from more than one source, those sources are listed in a drop-down menu. Possible sources are AWS Connections or public S3 buckets configured in the *CLC Workbench*, or configured in a *CLC Server* that the Workbench is connected to. A server icon (S) is shown beside sources available via a *CLC Server* (figure 3.7).

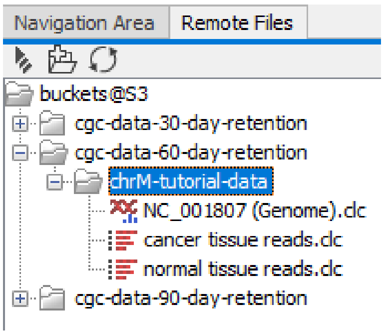


Figure 3.6: *AWS S3 buckets you have access to are available under the Remote Files tab.*

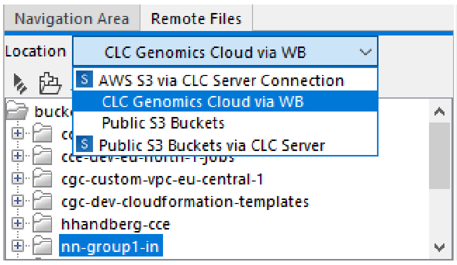


Figure 3.7: *This Workbench has a valid AWS Connection and is connected to a CLC Server with a valid AWS Connection. At least one public S3 bucket has been configured in the Workbench and in the CLC Server. The S3 buckets available from the selected source are listed in the Remote Files tab.*

Uploading data to AWS S3 using the Remote Files tab

To upload data from your Navigation Area to AWS S3, right-click on a folder in the Remote Files tab and choose the option **Upload to this folder** (figure 3.8).

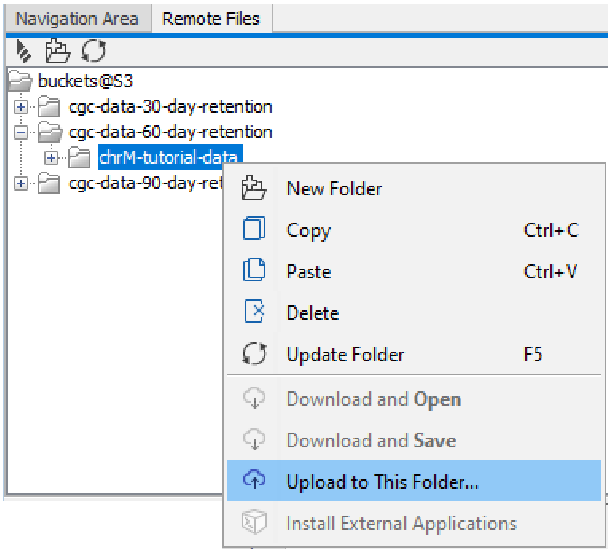
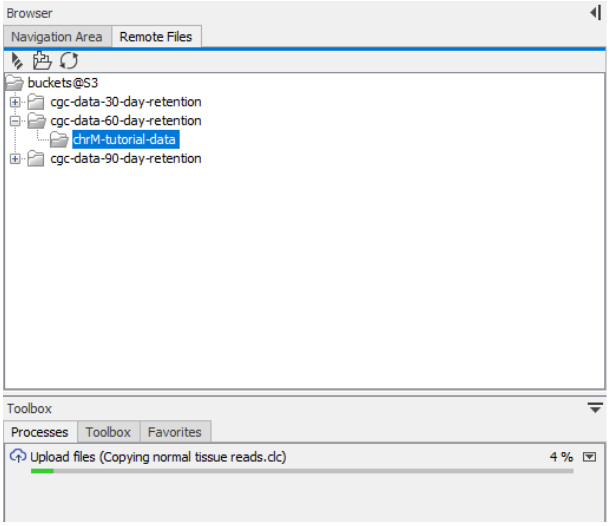


Figure 3.8: *To upload data from your Navigation Area to AWS S3, open the Remote Files tab, right-click on the folder you wish to upload data to, and select the option "Upload to This Folder".*

Upload is sequential. Information about the data upload is shown in the Processes tab, at the bottom left of the Workbench (figure 3.9).



Upload files (Copying normal tissue reads.clc) 4 %

Figure 3.9: *After choosing to upload data to S3, the progress of the upload is reported in the Processes tab.*

Downloading results via the Remote Files tab

Right-click on an element or elements under the Remote Files tab to download from AWS S3 (figure 3.10). When only a file or files are selected, options to **Download and Open** and **Download and Save** will be available. When a folder has been selected, only the **Download and Save** is available.

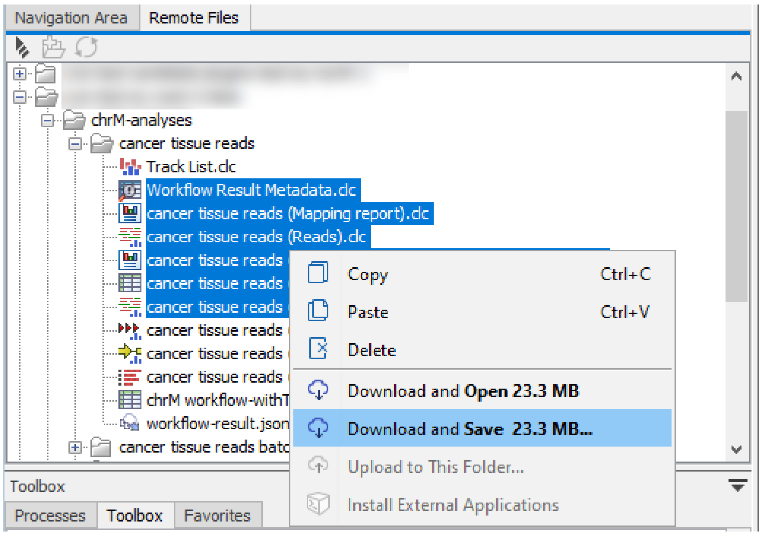


Figure 3.10: *Select folders and/or files in the Remote Files tab and right-click to reveal options for downloading that data.*

After choosing to download the data, you can choose to download it to a CLC File Location or to another area on your system.

To see all the outputs of a particular job that has been run on a *CLC Genomics Cloud* setup, double-click on a `workflow-result.json` file. All the results can then be downloaded and opened from that list, or individual elements can be selected and downloaded. The Execution Log is also available from this list (see figure 3.11).

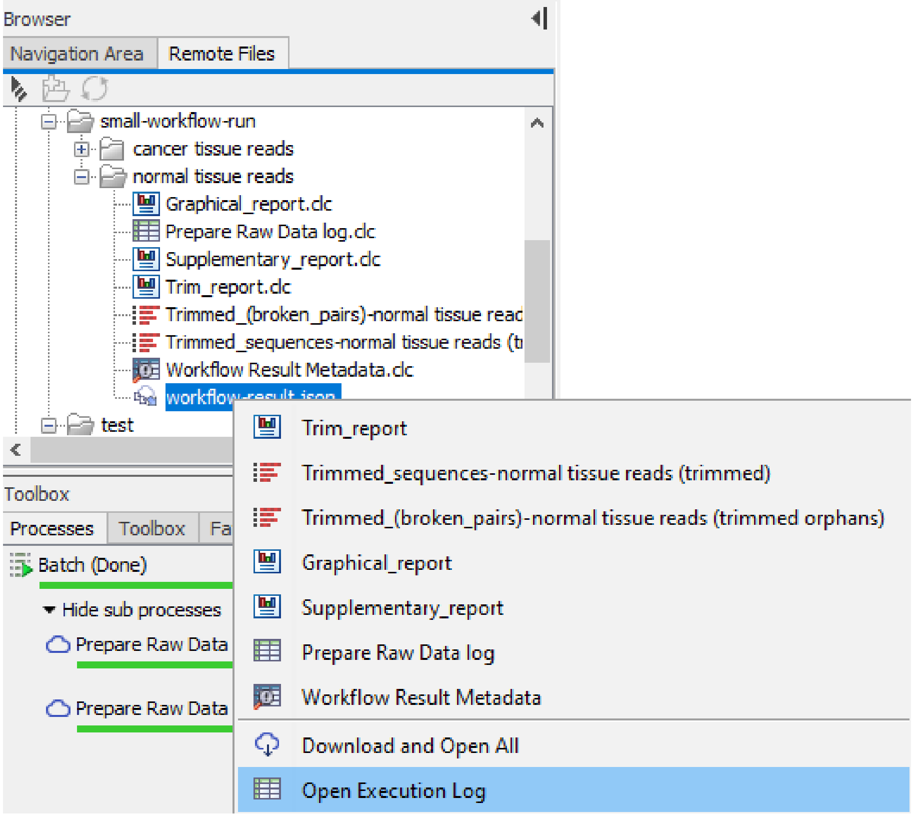


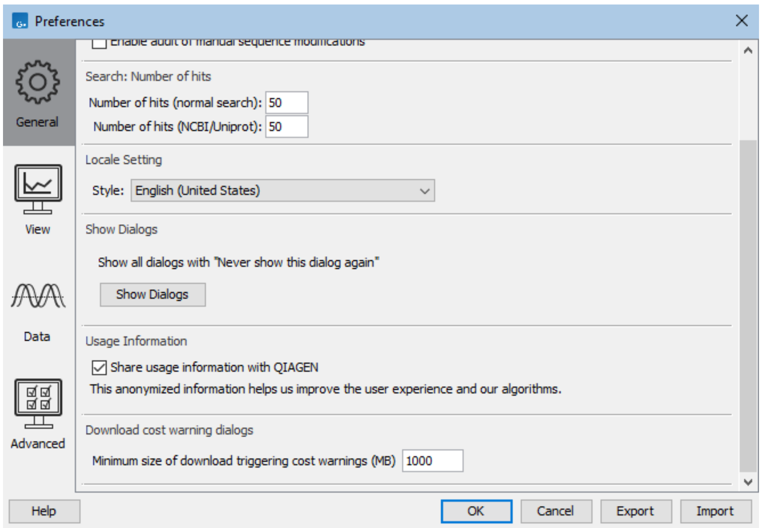
Figure 3.11: *Double-click on a workflow-result.json file in the Remote Files tab in the Workbench to reveal a list of all results from a job run in the cloud, as well as the Execution Log. All items can be downloaded and opened from this menu, or individual items can be selected and downloaded.*

If the Navigation Tools plugin is installed, bookmarks for items in the Remote Files tab can be made. Double-clicking on bookmarks for individual results files or folders opens the bookmarked items, as standard. Double-clicking on a bookmark for a workflow-result.json file reveals the same list of options as double-clicking on the workflow-result.json file in the Remote Files tab directly. Further details about bookmarks are provided in the Navigation Tools manual at https://resources.qiagenbioinformatics.com/manuals/navigationtools/current/index.php?manual=Introduction.html.

Note: AWS charges for downloading data from S3. By default, when the download size exceeds 1 GB, you are prompted for confirmation that you wish to proceed. The size required to trigger this warning can be changed in the General section of the Workbench Preferences (figure 3.12).

Downloading data using a URL

Pasting an URL into the Workbench Navigation Area will import the files, or a folder containing files, using Standard Import. File types are automatically detected. Thus, as well as CLC format files being available to view and work further with in the Workbench, other files in formats recognized by Standard Import will be imported as CLC format files, allowing them to be worked with in the Workbench. Further details about Standard Import are pro-



☐ Enable audit of manual sequence modifications

Search: Number of hits

Number of hits (normal search): 50

Number of hits (NCBI/Uniprot): 50

Locale Setting

Style: English (United States)

Show Dialogs

Show all dialogs with "Never show this dialog again"

Usage Information

☑ Share usage information with QIAGEN

This anonymized information helps us improve the user experience and our algorithms.

Download cost warning dialogs

Minimum size of download triggering cost warnings (MB): 1000

Figure 3.12: *The download size above which a cost warning dialog is shown can be adjusted in the Workbench Preferences. The default value is 1000 MB.*

vided at: https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Standard_import.html.

3.5 Accessing results from the Processes tab

Analysis results can be accessed via the menu available via the small triangle to the right of the process in the Processes tab (figure 3.13). Under the **Download and Open Results** option, you can select individual results to download and open, or you can download and open all results. When closing an element opened this way, you will be prompted for whether to save it. If it is not saved, it will need to be downloaded from AWS S3 again to be viewed.

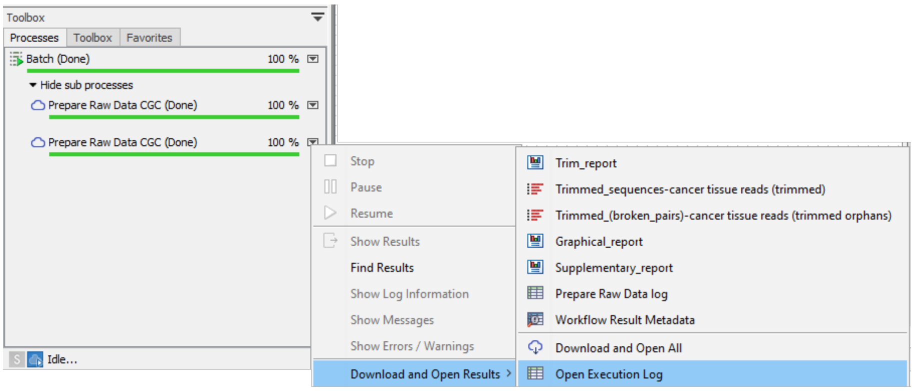


Figure 3.13: *Select "Download and Open Results" from the menu for a process to see a list of all the results as well as the Execution Log.*

This same list is available via the Remote Files tab by double-clicking on a `workflow-result.json` file.

When you select Find Results from under the Processes tab, the Remote Files tab will be opened, with the relevant folder opened and the outputs highlighted.

Note: When launching a job in Batch mode, a process labeled "Preparing N batch units" will be present, with N being the number of batch units. This process records the preparation of the individual jobs in the batch.

The Processes tab is described in more detail in section 7.1.

AWS charges for downloaded data from S3. See section 6.2 for further information about downloading data from AWS S3.

Chapter 4

Using a CLC Server to work on the cloud

Jobs can be submitted to the *CLC Genomics Cloud* via a *CLC Genomics Server*. In this chapter, we cover:

- Benefits of submitting to the cloud via a CLC Server (section 4.1)
- Cloud preset configuration (section 4.3)

Prerequisites for CLC Genomics Cloud access via a CLC Server

To submit jobs to a *CLC Genomics Cloud* setup via a *CLC Server*, the following is required:

- The Cloud Server Plugin has been installed on the *CLC Server*.
- An AWS Connection has been configured in the *CLC Server* using credentials for an AWS IAM user that can submit jobs to AWS Batch queues that are part of a *CLC Genomics Cloud* setup.
- At least one cloud preset has been configured.
- The *CLC Server* user is in a group with permission to use at least one cloud preset. (By default, all groups have access to all cloud presets and all AWS Connections.)

Installing server plugins is described in section 1.5.

4.1 Benefits of submitting jobs via a CLC Server

Working with a *CLC Server* brings several benefits when interacting with a *CLC Genomics Cloud* setup, including:

- **Audit trail** Jobs submitted to the cloud are logged, just as other jobs sent to the server. The audit log contains details of the job such as who submitted it, to which AWS Batch queue, whether the job succeeded and failed, etc. Further details are provided in section 7.2.

- **User management** A single set of AWS IAM credentials can be entered in an AWS Connection in the CLC Server, instead of configuring credentials in each individual CLC Workbench. All jobs sent to the cloud will use those AWS credentials. The CLC Server user that submitted a job is recorded in the server audit log. *CLC Server* user management supports connections to LDAP or Active Directory, as described at `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Authentication_options.html`.
- **Access control** Access to an AWS Connection can be limited to certain groups, allowing fine control over access to particular accounts or resources. In addition, access to AWS Batch queues is specified in CLC Genomics Cloud presets, configured in the *CLC Server* web administrative interface. Access to cloud presets can also be limited to specified groups.
- **Extend functionality using external applications** Non-CLC tools can be launched using CLC software by configuring them as *external applications*. Following the steps described in chapter 5, external applications can be launched directly from a *CLC Workbench*, from under the Tools menu, or in a workflow context, to run on the cloud.
- **Control over software versions used** When submitting jobs via a *CLC Server*, the version of the CLC software used for the cloud job is the same as the *CLC Server* version. When submitting jobs that include tools provided by a plugin, the plugin version used in the cloud is the same as that installed on the *CLC Server*. See section 8.2 for further details.
- **Close the CLC Workbench after job or data transfer has started** When launching analyses on the cloud using data from a CLC Server import/export area, you close down your CLC Workbench, or disconnect from the CLC Server, and transfers to S3 associated with the job (e.g. the workflow, parameter settings and any data held on the CLC Server or in import/export directories) will continue to be transferred. If submitting directly from a CLC Workbench, the process must proceed beyond data transfer steps before it can be closed.

4.2 AWS Connections in the CLC Server

Access to AWS accounts is configured under:

Configuration | **External data** | **AWS Connections**

Existing AWS Connections can be managed from under this tab and new AWS Connections created.

AWS Connections are used when:

- Accessing AWS S3 locations, to import data from or export data to.
- Submitting analyses to a *CLC Genomics Cloud* setup, if available on that AWS account.

All traffic to and from AWS is encrypted using a minimum of TLS version 1.2. AWS credentials entered are stored, obfuscated, in the server configuration files.

Creating a new AWS Connection

To create a new AWS Connection, click on the **Add AWS Connection...** button and provide the information requested:

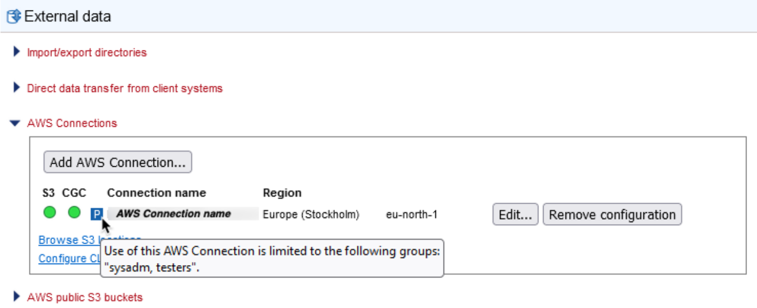
- **Connection name**: A short name of your choice, identifying the AWS account. This name will be shown as the name of the data location when importing data to or exporting data from Amazon S3.
- **AWS access key ID**: The access key ID for programmatic access for your AWS IAM user.
- **AWS access secret key**: The secret access key for programmatic access for your AWS IAM user.
- **AWS Region**: An AWS region. Select from the drop-down list.
- **AWS Partition**: The AWS partition for your account.

Information in the AWS Connections tab

Information about AWS Connections, including their status, is provided under the AWS Connections area (figure 4.1). Valid connections have a green indicator. Hovering the mouse cursor over the status reveals a tooltip containing further information about it.

Connections to a *CLC Genomics Cloud*, indicated in the CGC column, require the Cloud Server Plugin to be installed, and access to a CLC Genomics Cloud.

If permissions have been applied to an AWS Connection, a blue P symbol next to the connection name indicates this. Hovering the mouse cursor over this symbol reveals a tool tip with information about the groups that have access to that AWS Connection. Permissions can be set on AWS Connections under the **Global permissions** tab in the *CLC Server* web administrative interface.



Use of this AWS Connection is limited to the following groups: "sysadm, testers".

Figure 4.1: *Information about existing connections includes the name, region, whether permissions have been set and the connection status. Hovering the mouse cursor over the blue P symbol reveals a tooltip with details about the permissions that have been set. The green dot in the S3 column indicates a valid connection for accessing S3 buckets and the green dot in the CGC column indicates a valid connection to a CLC Genomics Cloud setup, meaning analyses can be submitted to run on AWS via this CLC Server.*

Clicking on the **Browse S3 locations** link opens the relevant tab under Element info. See section 4.4 for further details about this.

Configuring access to public S3 buckets is described at `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=AWS_S3_public_buckets.html`

4.3 Configuring cloud presets

When submitting a job to a *CLC Genomics Cloud* setup via a *CLC Server*, the submitter specifies a cloud preset to use. Each preset is configured with the name of a single AWS Batch queue. When a given preset is selected, jobs are sent to the corresponding AWS Batch queue.

Creating and editing cloud presets

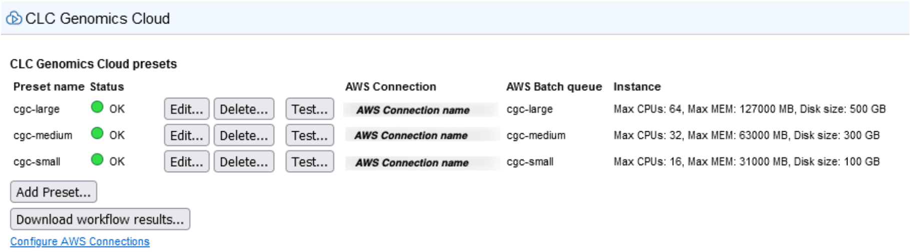
To view information about existing cloud presets or edit those presets, or to create new cloud presets, log into the *CLC Server* web administrative interface and go to:

Extensions ( ) | CLC Genomics Cloud ( )

New cloud presets can be added by clicking on the **Add Preset...** button. Existing cloud presets can be edited, tested, or deleted (figure 4.2). When creating or editing a cloud preset, the AWS Connection to use is selected from a drop-down list, and then the AWS Batch queue to use for that preset can be selected (figure 4.3).

Notes relating to cloud preset configurations:

- The preset name is used within the *CLC Server* web interface and is also the name presented to users via client software when they are preparing to launch an analysis.
- The list of AWS Connections is limited to those that the *CLC Server* user the administrator has logged in has permission to access. By default, all AWS Connections are available to all users. If permissions have been set, please ensure that the admin group is granted the access they need. (The CLC Server root user has access whether or not permissions have been applied.)
- The AWS Batch queues listed are those available to the AWS IAM user whose credentials are configured in the selected AWS Connection.

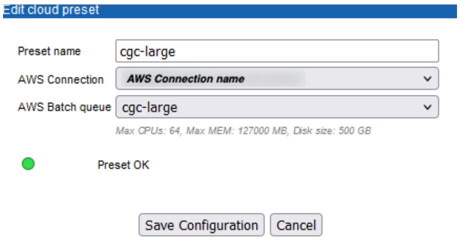


CLC Genomics Cloud presets

Preset name	Status	AWS Connection	AWS Batch queue	Instance
cgc-large	● OK	AWS Connection name	cgc-large	Max CPUs: 64, Max MEM: 127000 MB, Disk size: 500 GB
cgc-medium	● OK	AWS Connection name	cgc-medium	Max CPUs: 32, Max MEM: 63000 MB, Disk size: 300 GB
cgc-small	● OK	AWS Connection name	cgc-small	Max CPUs: 16, Max MEM: 31000 MB, Disk size: 100 GB

Figure 4.2: *Cloud presets can be created, managed and tested under the CLC Genomics Cloud tab. This requires the Cloud Server Plugin to be installed and at least one AWS Connection where the IAM user has access to a CLC Genomics Cloud setup.*

An AWS Connection must have been configured before cloud presets can be configured.



Preset name: cgc-large

AWS Connection: AWS Connection name

AWS Batch queue: cgc-large

Max CPUs: 64, Max MEM: 127000 MB, Disk size: 500 GB

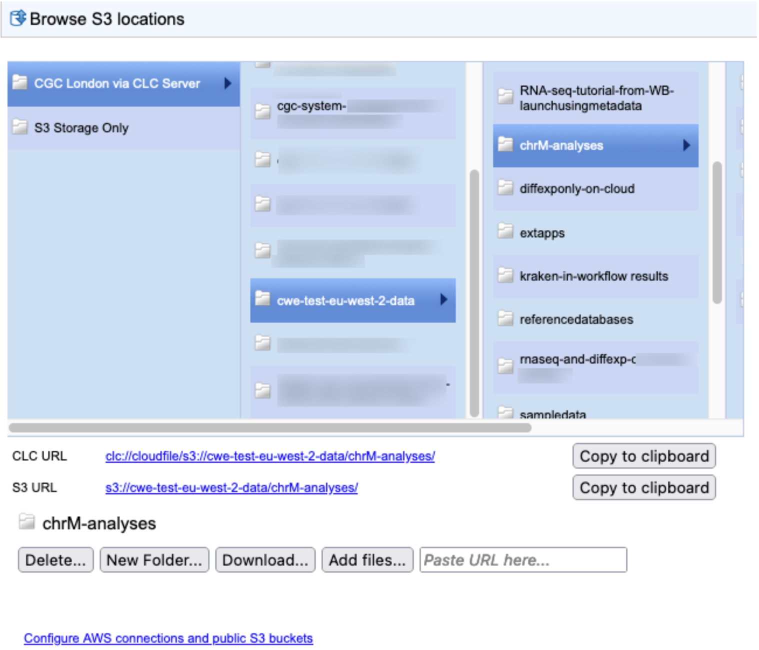
● Preset OK

Figure 4.3: *When editing or adding cloud presets, the AWS Connection and the relevant AWS Batch queue are specified. Information about the status of the preset is provided. A green dot indicates that the connection is valid.*

4.4 Browse AWS S3 locations

The contents of AWS S3 buckets you have access to can be browsed by going to:

Element info ( ) | **Browse S3 locations** ( ) .



CLC URL: clc://cloudfile/s3://cwe-test-eu-west-2-data/chrM-analyses/

S3 URL: s3://cwe-test-eu-west-2-data/chrM-analyses/

Figure 4.4: *Data can be uploaded, downloaded, deleted and new folders created using functionality in the "Browse S3 locations tab". A folder in an S3 bucket that is being used to store sample data has been selected. URLs for that folder are provided under the file browser. Here, an admin user is logged into the CLC Server, so the CGC system bucket is visible.*

When an item in an S3 bucket is selected, URLs to that item are shown below the file listing area and relevant options are enabled (figure 4.4). These URLs can be copied using the **Copy to clipboard** button. They can be useful when copying data to server import/export directories, when sharing the location of files with others, and when specifying inputs and locations to save to using the *CLC Server Command Line Tools*.

The actions available are:

- **Delete files or folders** If a folder has been selected, clicking on the **Delete...** button opens a dialog offering the option to delete the folder and its contents, or to specify individual files from that folder to be deleted. You are asked for confirmation of your intention to proceed with deletion.

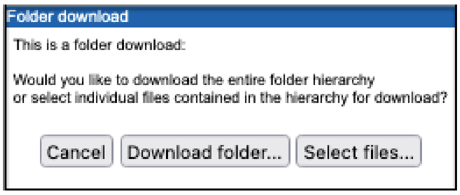
- **Add a new folder** Click on the **New Folder...** button to create a new folder within the folder selected in the browsing area.
- **Download files or folders** Data can be downloaded to a CLC File System Location or to an import/export directory.

  If a folder has been selected, clicking on the **Download...** button opens a dialog with the options **Download folder...**, to download the folder and its contents, and **Select files...**, to select individual files in that folder for download (figure 4.5).

  When selecting individual files, the files in the selected folder are listed, along with their sizes, and the total download size of the selected files (figure 4.6). After download, a list of the downloaded files is presented under the Results area at the bottom of that dialog.

  Download starts when the location to download to has been selected.

  CLC data downloaded to a CLC location, can be opened using a *CLC Workbench* connected to the *CLC Server* and used in downstream analyses.



This is a folder download:

Would you like to download the entire folder hierarchy
or select individual files contained in the hierarchy for download?

Figure 4.5: *If a folder is selected when the Download... button is clicked, you get the option to download the folder (and its contents) or to select files from within the folder to download.*

  Note: AWS charges for downloading data from AWS S3.

- **Upload data**

  Data from a CLC File System Location, from an import/export directory or from AWS S3 can be uploaded to the selected folder:

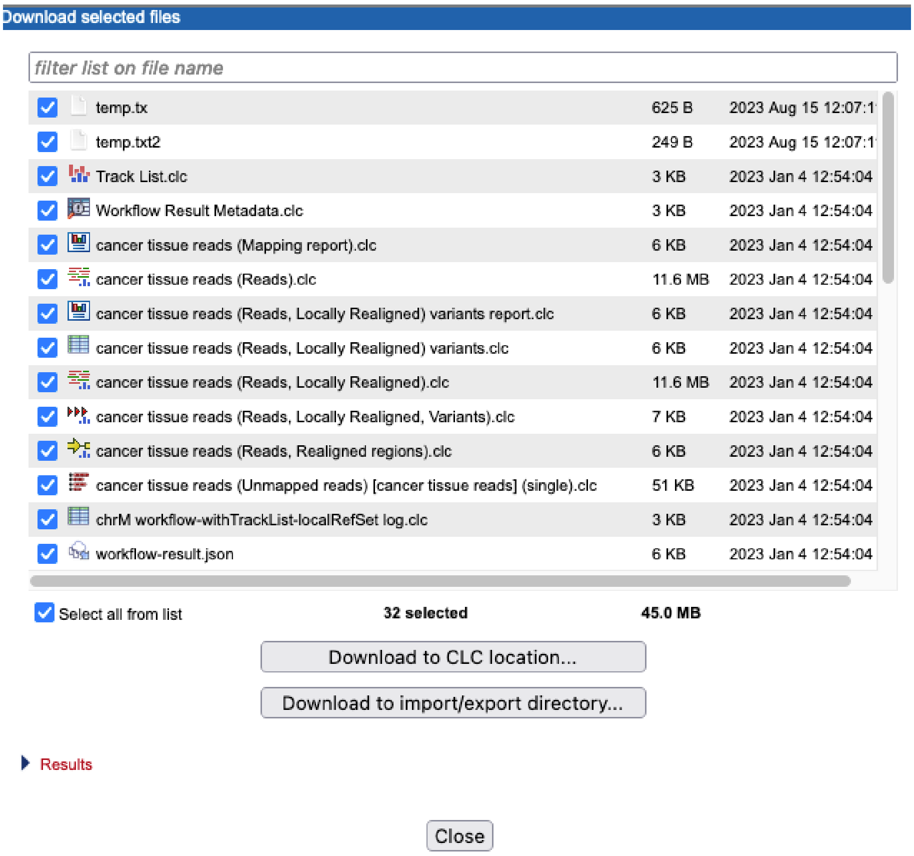
  There are two ways to indicate the data to upload:

  - **Using a graphical file chooser** Click on the **Add files...** button to select files and folders using a graphical file browser. The first step will be to specify the source of the data to upload (figure4.7).

    A graphical file browser will open allowing selection of the files and/or folders to upload.
  - **Providing a URL** URLs for data in a CLC Server File System Location are shown when browsing CLC data in the web client under **Element info** ( ) | **Info** ( ) and can also be obtained by using the Copy function in the Navigation Area of a Workbench.

    URLs for data in an import/export area are shown when browsing import/export directories under **Element info** ( ) | **Browse server import/export directories** ( ).

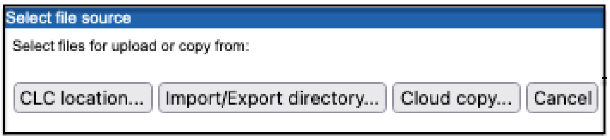
    URLs for data in AWS S3 buckets are available when browsing using the web client, as described on this page.



☑	temp.tx	625 B	2023 Aug 15 12:07:1
☑	temp.txt2	249 B	2023 Aug 15 12:07:1
☑	Track List.clc	3 KB	2023 Jan 4 12:54:04
☑	Workflow Result Metadata.clc	3 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Mapping report).clc	6 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads).clc	11.6 MB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads, Locally Realigned) variants report.clc	6 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads, Locally Realigned) variants.clc	6 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads, Locally Realigned).clc	11.6 MB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads, Locally Realigned, Variants).clc	7 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Reads, Realigned regions).clc	6 KB	2023 Jan 4 12:54:04
☑	cancer tissue reads (Unmapped reads) [cancer tissue reads] (single).clc	51 KB	2023 Jan 4 12:54:04
☑	chrM workflow-withTrackList-localRefSet log.clc	3 KB	2023 Jan 4 12:54:04
☑	workflow-result.json	6 KB	2023 Jan 4 12:54:04

☑ Select all from list 32 selected 45.0 MB

Figure 4.6: *When downloading files from S3, a list of the selected files and the total size of the download is displayed. You will be prompted for a specific location to download to.*



Select files for upload or copy from:

Figure 4.7: *Choose the source of data to upload to AWS S3.*

The actions that can be taken in practice depend on the permissions for accessing the selected AWS S3 bucket and settings in the *CLC Server* for the user logged in.

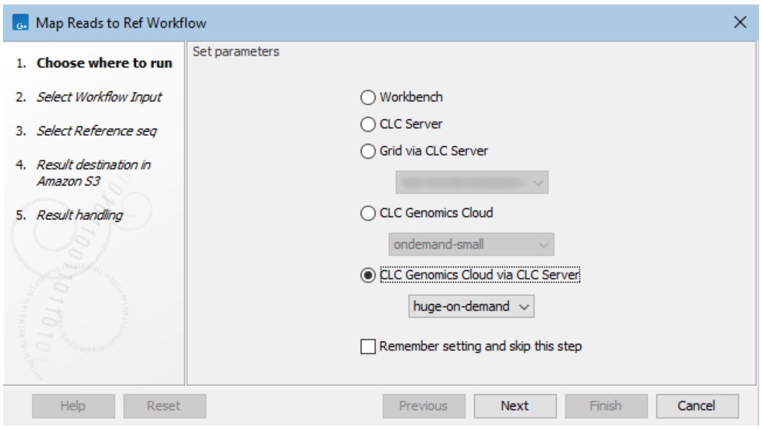
Additional notes

- If the message "No active S3 locations found." is visible in the "Browse S3 locations" tab, it means that either no AWS Connections or AWS S3 public buckets have been configured, or that there is no access for the user logged in.
- If you are logged into the *CLC Server* web client as an administrative user, then in addition to S3 buckets that can be used to store input data and results, the CGC system bucket will be listed (figure 4.4). This bucket has a name starting with `cgc-system`. It is used for storing system files. It is not intended for storing sample data or results. It can be browsed, but there is generally no need to do so. When logged in as a non-admin user,

the CGC system bucket will not listed by name and cannot be accessed. (Note: the CGC system bucket is not listed or accessible via CLC Workbenches, whether logged into the CLC Server as an admin or non-admin user.)

4.5 Submitting workflows to the cloud via a CLC Server using a CLC Workbench

To submit jobs to a *CLC Genomics Cloud* setup via a *CLC Server*, choose the launch wizard option "CLC Genomics Cloud via CLC Server" (figure 4.8).



○ Workbench
○ CLC Server
○ Grid via CLC Server
○ CLC Genomics Cloud
ondemand-small
● CLC Genomics Cloud via CLC Server
huge-on-demand
☐ Remember setting and skip this step

Figure 4.8: *When a CLC Workbench with a licensed CLC Cloud Module installed is connected to a CLC Server configured for access to a CLC Genomics Cloud, the option to submit jobs to the cloud via the server is enabled.*

4.6 Submitting workflows to the cloud using the CLC Server Command Line Tools

Submitting workflows to the cloud

Workflows installed on a *CLC Genomics Server* can be submitted to run in the cloud using the *CLC Server Command Line Tools*. General information about launching workflows using the *CLC Server Command Line Tools* is at https://resources.qiagenbioinformatics.com/manuals/clcservercommandlinetools/current/index.php?manual=Launching_workflows.html. That information focuses on submitting jobs for execution on a *CLC Genomics Server*, but most of it also applies to workflows submitted for execution on the cloud. This section provides cloud-specific details.

Specifying a cloud preset

The cloud preset to use must be specified when submitting workflows to run on the cloud. The name of the cloud preset should be supplied as the value for the **-L** option.

To see the list of cloud presets available, run the `clcserver` command with no arguments or with an incomplete set of arguments, as described on https://resources.qiagenbioinformatics.com/manuals/clcservercommandlinetools/current/index.php?manual=Basic_usage.html.

Information about configuring cloud presets on a *CLC Server* is in section 4.3.

Specifying input data for analyses

CLC format data in CLC Server File System Locations or in remote locations accessible via http, https, or S3 URL can be provided as inputs to workflows[1].

Data in other formats can be supplied as input by using on-the-fly import. For example, using on-the-fly import, FASTQ sequence files would be imported as the first step in the workflow, avoiding the need for running a specific import command before running the workflow.

General information about input data for analyses run on the cloud is provided in section 6.1.

Further details about providing input data to analyses using the *CLC Server Command Line Tools* is at `https://resources.qiagenbioinformatics.com/manuals/clcservercommandlinetools/current/index.php?manual=Providing_input_data_analyses_on_CLC_Server.html`.

Specifying where results should be saved

Results generated using workflows run on the cloud are saved to AWS S3. The location to save results to is specified using an S3 URL as the value for the relevant parameter.

General information about specifying where results should be saved when using the *CLC Server Command Line Tools* is provided at `https://resources.qiagenbioinformatics.com/manuals/clcservercommandlinetools/current/index.php?manual=Saving_workflow_outputs_exporting_results.html`.

Accessing AWS CloudWatch logs via the command line

The *CLC Server Command Line Tools* command `-A cgc_read_aws_logs` supports the retrieval of AWS CloudWatch logs for jobs run on a *CLC Genomics Cloud*.

The messages returned from jobs run on the cloud include the information needed to access the AWS CloudWatch log for that job. The AWS CloudWatch information retrieved is the same as that returned when the "Execution Log" is opened in the CLC Workbench, either via the Processes tab or via options under the Remote Files tab, as described in section or chapter 3.5.

---

[1] Support for http, https and S3 URLs for directly specifying files in remote locations, i.e. not needing to specify the location as a clccloudfile, was introduced in *CLC Genomics Server* **23.0.3** and Cloud Server Plugin **23.0.1**, as was the ability to supply CLC format files in remote locations directly as input, without needing to use on-the-fly import.

Chapter 5

Integrating third party tools into CLC software

Third party applications and scripts can be integrated with CLC software as *"external applications"*.

When included in a workflow, external applications on a *CLC Server* can be submitted for execution on the *CLC Server* or in the cloud using a *CLC Workbench* or using the *CLC Server Command Line Tools*.

Workflows containing external applications can also be submitted for execution on the cloud directly from a *CLC Workbench*, without connecting to a *CLC Server*. This requires the external applications to be installed on the *CLC Workbench* and then added to the workflow. Such workflows can only be executed on the cloud.

Related documentation links

- Creating and configuring external applications, done in the web client of the *CLC Server*: `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=External_applications.html`

  By default, only administrative users can configure external applications, but permissions can be granted to others. See "Web admin access" on `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Controlling_access_server_server_tasks_external_data.html`.

- Adding an external application to a workflow, done in the Workflow Editor of a *CLC Workbench*: `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=External_applications_in_workflows.html`.

  Links to general information about creating and editing workflows are provided at the bottom of that page.

- Submitting workflows to the cloud from a *CLC Workbench* via a *CLC Server*: section 4.5.
- Submitting workflows to the cloud using the *CLC Server Command Line Tools*: section 4.6
- Installing external applications on a *CLC Workbench*: section 5.1
- Submitting workflows to the cloud from a *CLC Workbench* directly: section 3.3

Extra requirements for external applications to be run on the cloud

- To run an external application on the cloud, the resources referred to by it must be available on the cloud.

  For containerized external applications that can be successfully run on a *CLC Server*, this requirement will generally be met.

  For standard external applications, the tools referred to in the configuration, including any referred to in an included script in that configuration, need to be available in the Linux image used when the job is run on AWS. See appendix section 8.2 for information about the compute resources in a *CLC Genomics Cloud* setup.
- When launching via a *CLC Server*, the external application is present on the *CLC Server* and accessible to the user launching the workflow.
- When launching directly from a *CLC Workbench*, the external application has been installed on the *CLC Workbench* and then included in the workflow (see section 5.1).

5.1 Installing external applications in a CLC Workbench

Submitting a workflow containing external applications to the cloud directly from the *CLC Workbench* is useful in situations where the person who should run the workflow does not have access to a *CLC Server* where the external applications are available. To do this, the external applications in the workflow must be available from the *CLC Workbench*. This involves the following steps:

1. **Export and share the external application**

   From the *CLC Server* web interface, export the external application configuration(s) to an AWS S3 location accessible from the *CLC Workbench* that the external application will be installed on.

   Exporting external application configurations is described at https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Import_export_external_application_configurations.html.

   More than one external application can be exported to a single file.

   To take these actions, you must be logged into the *CLC Server* as an administrator or as a user with rights to configure external applications.

   Some additional considerations when making external applications available this way are listed further down the page.

2. **Install the external applications in the CLC Workbench**

   In a *CLC Workbench* with the CLC Cloud Module installed and a license for that module present:

   (a) Navigate to the relevant AWS S3 location using the Remote Files tab (figure 5.1).

   (b) Select one or more external application configuration files, right-click, and choose the option **Install External Applications** (figure 5.2).

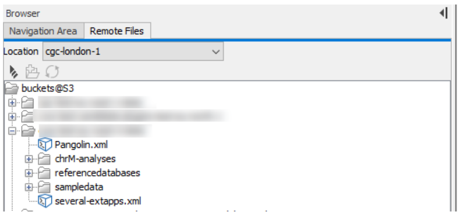


Figure 5.1: *Two separate external application configuration files, Pangolin.xml and several-extapps.xml, have been exported from a CLC Server to an AWS S3 location that is accessible from this CLC Workbench.*

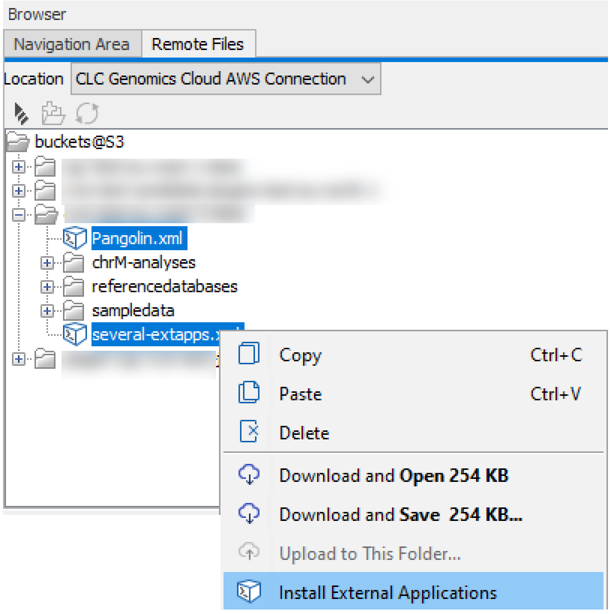
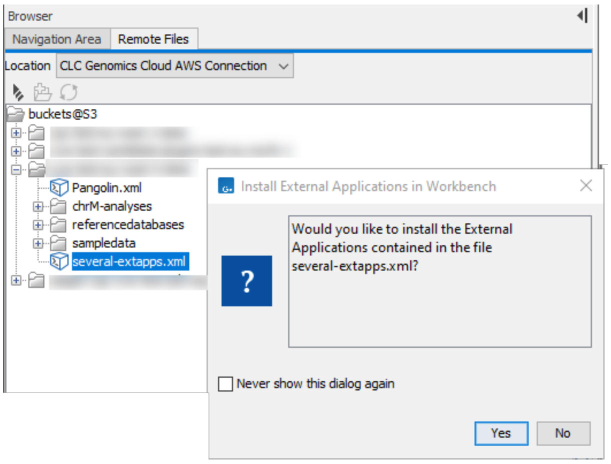


Figure 5.2: *Select one or more external applications in the Remote Files tab, right click, and choose Install External Applications.*

Alternatively, double-click on one of the external applications files in AWS S3 and confirm that you wish to proceed (figure 5.3).

If a file contains configurations for multiple external applications, all of them will be installed.



Would you like to install the External Applications contained in the file several-extapps.xml?

☐ Never show this dialog again

Figure 5.3: *Double-click on an external application configuration file in the Remote Files tab, and confirm you wish to install the external application(s) defined in that XML file.*

External applications installed this way are placed in the **External Applications Cloud** folder under the Tools menu (figure 5.4). These external applications can be included in a workflow and such workflows submitted to run on the cloud directly from the *CLC Workbench* (see section 3.3).

Note: The organization of the tools in the External Applications folder depends its configuration. This is under the control of your server administrator.

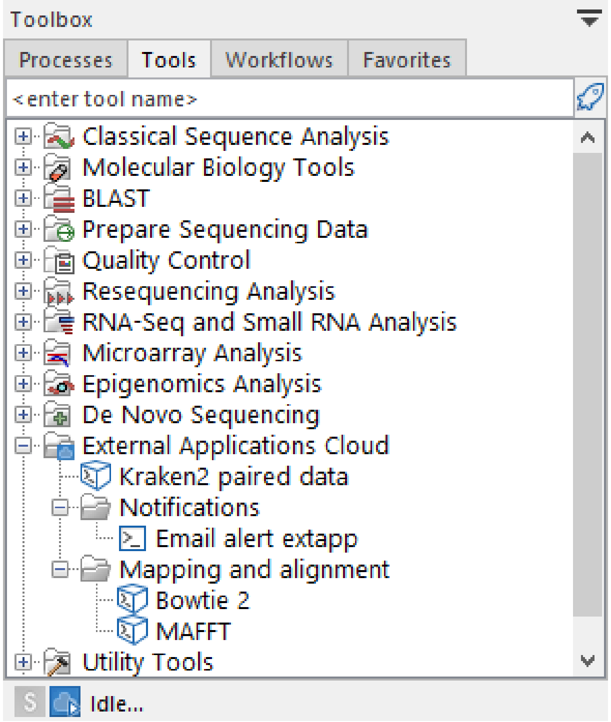


Figure 5.4: *External applications in the External Applications Cloud folder under the Tools menu can be included in workflows to be run on the cloud.*

Note: External applications cannot be run directly on the *CLC Workbench*, within or outside of a workflow.

Some considerations when making external applications available this way:

- Once an external application is installed on the *CLC Workbench*, the external application author and *CLC Server* administrator have no control over its use.
- Updated versions of an external application are not automatically deployed. The steps in the section above must be repeated to install new versions of external applications in the Workbench.
- If the external application installed in the *CLC Workbench* is a newer version than that included in an existing workflow, then that workflow must updated. Updating workflows in a *CLC Workbench* is described on `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Updating_workflows.html`.
- In a *CLC Workbench*, the same external application cannot be available via the *CLC Server* and present in the *CLC Workbench* at the same time.

Uninstalling a Workbench-installed external application

An external application can be uninstalled from the *CLC Workbench* by locating it in the Tools menu in the Toolbox as the bottom, left side of the Workbench, right-clicking on its name, and choosing **Uninstall External Application** from the menu that appears (figure 5.5).

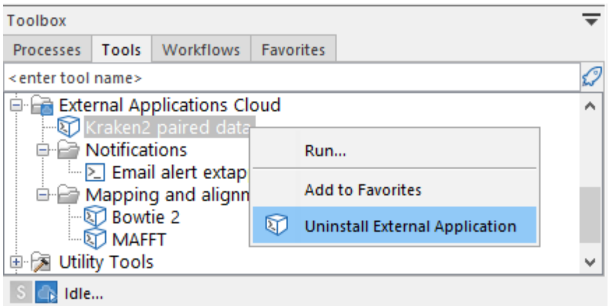


Figure 5.5: *Uninstall an external application by right-clicking on its name under the Tools menu in the Toolbox and selecting the option to uninstall it from the menu.*

Chapter 6

General information about inputs and results

This chapter covers general considerations about the location of input data, including reference data, and working with results of CLC analyses run on the cloud.

6.1 General information about input data for cloud analyses

Cloud-based analyses use input data in AWS S3. Data not already in S3 is transferred from the local system to the cloud at the start of an analysis job. Thus, it is often most efficient to use data already in AWS S3 for these analyses.

CLC files can be uploaded to AWS S3 directly via the *CLC Workbench*, as described in section 3.4.

Raw data, for example, FASTQ files, can be uploaded to AWS S3 using third party tools, or via the web interface of a *CLC Server*, as described at `https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Browse_AWS_S3_locations.html`.

QIAGEN reference data is already available on AWS S3, so no transfer of this data is needed. This is discussed further in section 6.1.1.

The few situations where using locally held data is the only option are described in section 6.1.2.

6.1.1 Reference data for analyses on the cloud

When running workflows configured to use QIAGEN reference data, no transfer of that data will take place, and there is no need to explicitly copy it to your own AWS S3 account. Under certain conditions, no local copy of this reference data is needed either. This is described in more detail later in this section.

In cases where other reference data will be used, the following steps are recommended:

- Connect Input elements to all input channels that require reference data (figure 6.1).

  When a workflow Input element is connected, you are presented with the option of using on-the-fly import. This allows you to select data in an AWS S3 bucket (figure 6.2).
- Upload reference data to AWS S3 before launching analyses.

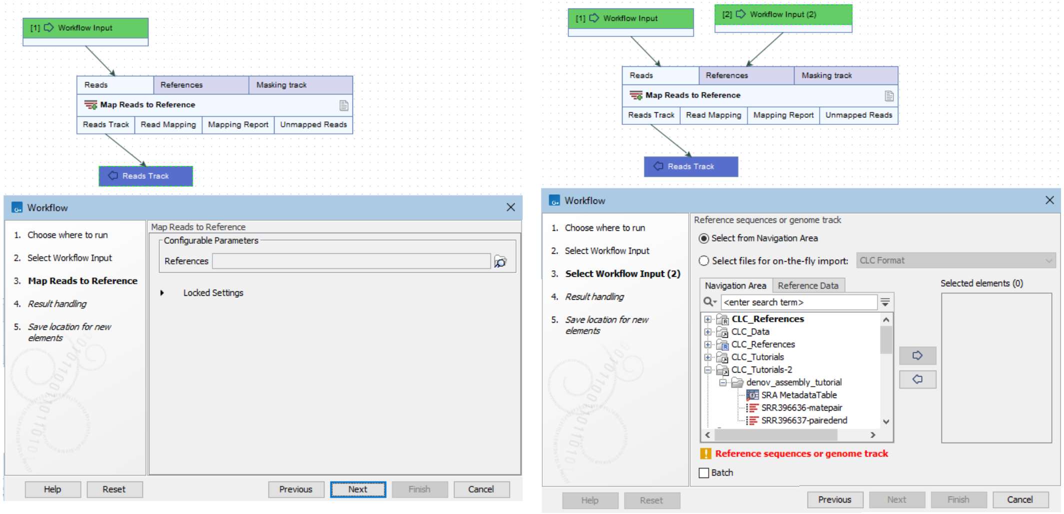
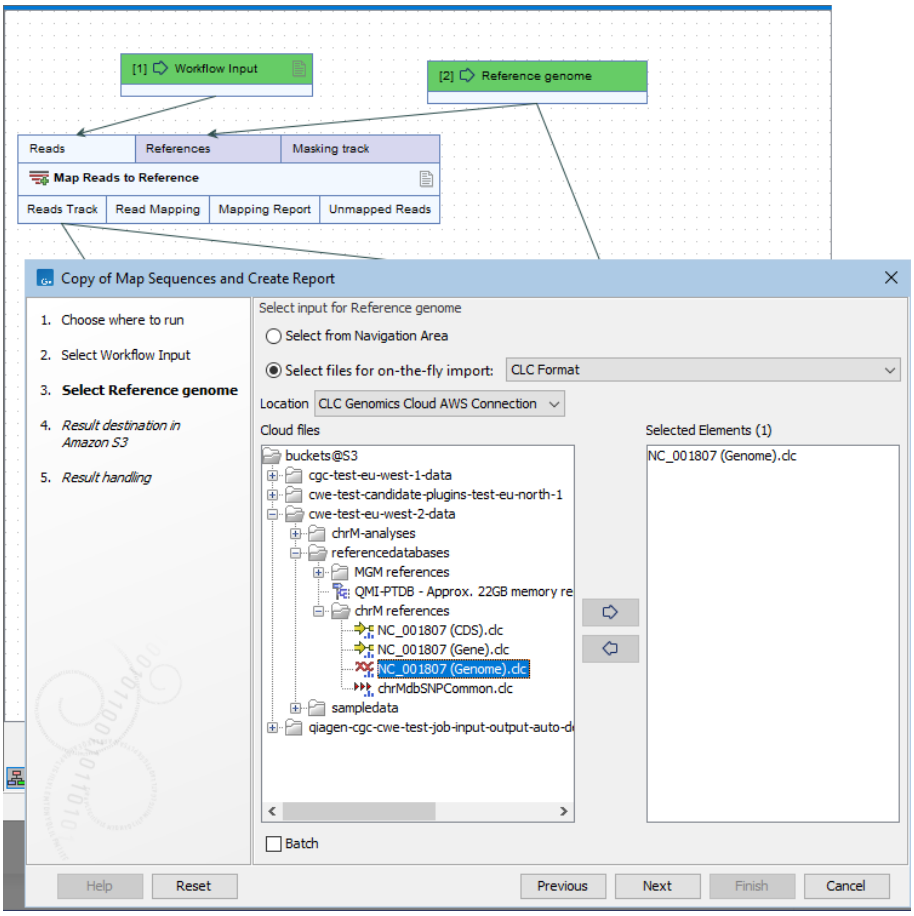


Figure 6.1: *Using the workflow on the left, data for the References field can only be selected from a CLC Location. The workflow on the right has an Input element connected to the References input channel. Using that workflow, files can be selected from an AWS S3 bucket, or from other accessible places, including CLC Locations.*



Select input for Reference genome

○ Select from Navigation Area

● Select files for on-the-fly import: CLC Format

Location CLC Genomics Cloud AWS Connection

Cloud files

- buckets@S3
  - cgc-test-eu-west-1-data
  - cwe-test-candidate-plugins-test-eu-north-1
  - cwe-test-eu-west-2-data
    - chrM-analyses
    - referencedatabases
      - MGM references
      - QMI-PTDB - Approx. 22GB memory re
      - chrM references
        - NC_001807 (CDS).clc
        - NC_001807 (Gene).clc
        - NC_001807 (Genome).clc
        - chrMdbSNPCommon.clc
    - sampledata
  - qiagen-cgc-cwe-test-job-input-output-auto-d

Selected Elements (1)

NC_001807 (Genome).clc

☐ Batch

Figure 6.2: *An Input element is connected to the References input channel. On-the-fly import of a CLC format file has been specified by selected the "Select files for on-the-fly import" option and "CLC Format" from the drop-down list of formats. The relevant AWS Connection has been selected from the drop-down list of locations. A CLC file was then selected for use as the reference genome.*

QIAGEN reference data in workflows

QIAGEN reference data elements[1] are already present in AWS S3 (figure 6.3) and thus do not need to be uploaded to your own S3 bucket when running workflows configured to refer to them. This includes many of the template workflows delivered with the software, and thus also workflows derived from those.

When the conditions listed below are met, there is also no need for a local copy of QIAGEN reference data when launching workflows to run on the cloud.

- All reference data parameters must be configured with a single QIAGEN reference data element and/or configured with a workflow role that is specified in one or more QIAGEN reference sets.

  Configuring workflows with roles for reference data is described at `https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Configuring_`

---

[1] "QIAGEN reference data" refers to data sets or data elements provided by QIAGEN, available from under the **QIAGEN references** tab of the Reference Data Manager. Template workflows, delivered with the software, are commonly configured to use QIAGEN reference data.

`Workflow_Input_elements.html`.

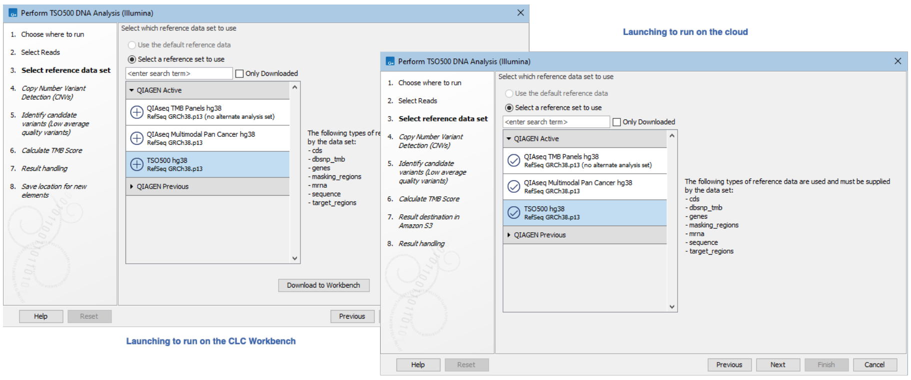
- All workflow parameters referring to QIAGEN reference data are locked.

  Where reference data elements need to be specified when launching, those elements, by definition, must be unlocked, and so a locally accessible copy of the relevant reference data will be needed. This is the case, for example, for many QIAseq analysis workflows provided by the *Biomedical Genomics Analysis* plugin, where there are drop-down menus for selecting target region and target primer sets. If QIAGEN reference elements are selected, they are not copied to S3 from your system. Instead, the copy provided by QIAGEN in AWS S3 will be used for analyses run on the cloud.

- Workflows are being submitted via a *CLC Workbench*.

  When submitting workflows that make use of QIAGEN reference data via a *CLC Server*, the reference data elements must be present in the *CLC Server CLC_References* location. Any QIAGEN reference elements selected will, however, not be copied to S3 from your system. The copy provided in AWS S3 by QIAGEN will be used for analyses run on the cloud.

Note: To view track lists that refer to reference data elements, those elements must be available locally.



The following types of reference data are used and must be supplied by the data set:
- cds
- dbsnp_tmb
- genes
- masking_regions
- mrna
- sequence
- target_regions

Figure 6.3: *QIAGEN Reference Sets do not need to be available locally (left hand image) for them to be available when launching a workflow to run on the cloud (right hand image).*

6.1.2 Using locally held data for analyses on AWS S3

Two considerations when using data stored locally for analysis on AWS S3 are:

1. Transfer of data from a non-cloud location, such as from CLC Locations, from local directories, or from CLC Server import/export directories, takes time, and when submitting jobs from the *CLC Workbench*, any necessary data transfer must be complete before the software can be closed.
2. Data uploaded to AWS S3 as part of job submission is not saved to AWS S3 for subsequent use. Later analyses using the same input data would involve another transfer of that data to the cloud.

Cases where locally held data must be used:

- Launching a tool from the *CLC Workbench* Tools menu to run on a CLC Genomics Cloud[2]. This tool will be wrapped in a workflow for you, before the job is sent to the cloud. When launching analyses this way, you can only select data held in CLC Locations. Thus, the first step when a tool has been launched will always be to upload the input data to AWS S3. Launching tools can be useful when running small tests, but for analysis of large data sets, or where data needs to be imported, we recommend creating a workflow containing the tool of interest and launching that instead.
- When specifying reference data elements in certain workflow designs, as described in section 6.1.1
- Running analyses that require a CLC Metadata Table with data associated to it using a workflow that does not include an Iterate element. An example would be a workflow with the Differential Expression tool at the top of the workflow.

Workflows containing Differential Expression can be run using data already on AWS S3 when that workflow contains an Iterate element upstream. An example illustrating this is the **RNA-Seq and Differential Gene Expression Analysis** template workflow, delivered with the *CLC Genomics Workbench*, described at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=RNA_Seq_Differential_Gene_Expression_Analysis_workflow.html. If the batch units for the the RNA-Seq analysis section of that workflow are defined using metadata, that same metadata will be used as input to the Differential Expression step.

6.2 General information about outputs from cloud analyses

All results generated on the cloud are stored on the cloud. To view results in a *CLC Workbench*, the data must be downloaded to a local CLC location. This is easily done via a *CLC Workbench* via the Remote Files tab, described in section 3.4, and via the Processes tab, described in section 3.5.

AWS charges for downloading data. In CLC Workbenches, when the download size exceeds 1 GB, you are prompted for confirmation that you wish to proceed. The size required to trigger this warning can be changed in the General section of the Workbench Preferences (figure 6.4). When downloading via the CLC Server web interface, a window pops up listing the selected files and providing the total download size. After selecting whether to download to a CLC location or to a CLC Server import/export area, you are prompted to selected a specific location to download to. For details, see https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Browse_AWS_S3_locations.html.

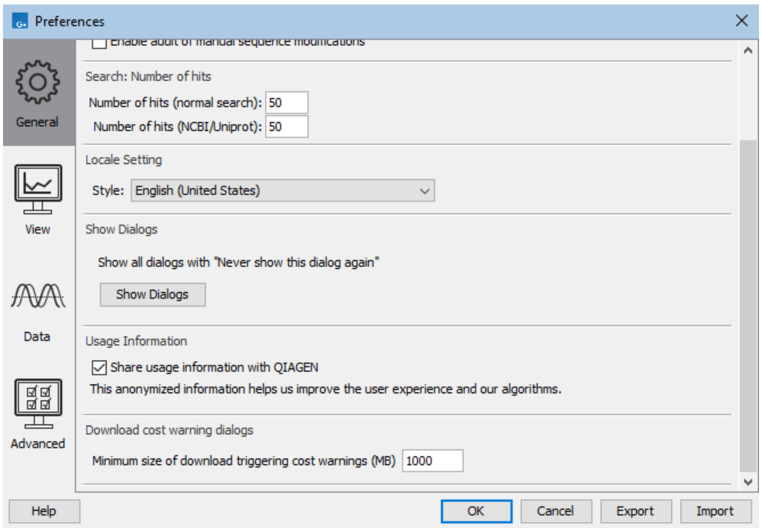
For downloading large amounts of data, we recommend using dedicated software for interacting with AWS S3. CLC data downloaded using external software can be imported into the *CLC Workbench* using Standard Import, described at https://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/current/index.php?manual=Standard_import.html.

Download all contained tracks when downloading track lists to view

Track lists are small elements that refer to tracks. When downloading track lists to view, the track list *and* all the tracks contained must be downloaded *at the same time*.

---

[2] Only tools that can be used within workflows can be submitted to run on a CLC Genomics Cloud. Tools that are not workflow-enabled cannot be run on the cloud.



☐ Enable audit of manual sequence modifications

Search: Number of hits

Number of hits (normal search): 50

Number of hits (NCBI/Uniprot): 50

Locale Setting

Style: English (United States)

Show Dialogs

Show all dialogs with "Never show this dialog again"

Usage Information

☑ Share usage information with QIAGEN

This anonymized information helps us improve the user experience and our algorithms.

Download cost warning dialogs

Minimum size of download triggering cost warnings (MB) 1000

Figure 6.4: *Download cost warning dialogs are shown whenever the size of a download from AWS S3 exceeds the value configured in the Workbench Preferences. By default, the value is set to 1000 MB.*

If this is not done, one or more of the tracks in the track list will not be resolved. The relevant links between track lists and the tracks referred to cannot be made if these elements are downloaded separately.

Chapter 7

Monitoring and reviewing CLC cloud jobs

CLC jobs run on the cloud are monitored using CloudWatch. Information about jobs can be found via CLC software as well as the AWS Console.

Key files for investigating cloud jobs are:

- **workflow-result.json** This file contains a list of the outputs from an analysis and accessing it via a *CLC Workbench* or *CLC Server*.
- The analysis log This is the usual source of information about analyses, where the focus is on the analysis steps run. This log provides information useful when investigating problems with the analysis itself. The filename include the name of the analysis followed by "log", for example, "Prepare Raw Data log" (figure 7.2).
- **Execution Log** This file contains the log stream information from Cloud Watch for the job. This file can be opened from the list provided in the Processes tab (figure 7.1), or by double-clicking on the `workflow-result.json` file in the Remote Files tab (figure 7.2). For very long logs, the first 10,000 entries from the log stream are included, as well as a few entries from the end of the log stream. If the full log is needed, it can be accessed directly in AWS CloudWatch.

  The Execution Log can also be retrieved, as long as the information is still available on AWS.

  The Execution Log is primarily for troubleshooting problems with a job on the cloud.

In this chapter, we describe monitoring jobs using:

- A CLC Workbench: section 7.1
- A CLC Server: section 7.2
- The AWS Console: section 7.3

7.1 Using a CLC Workbench to monitor and review cloud jobs

Information about the status of queued, running and completed jobs submitted from a *CLC Workbench* to the cloud, either directly, or via a *CLC Server*, appears in the **Processes tab** in the lower left side of the Workbench. When the CLC Workbench is closed and then re-opened, AWS Batch is queried, and processes for jobs are listed and updated with their progress. Any cloud job that has not previously been actively cleared from the process list will be shown.

When jobs have completed, results can be found, or downloaded and opened, using options in the menu found under the small icon to the right of the process. The standard log for an analysis (e.g. "Prepare Raw Data log" in the figures below) and the Execution Log are available from this menu (figure 7.1).

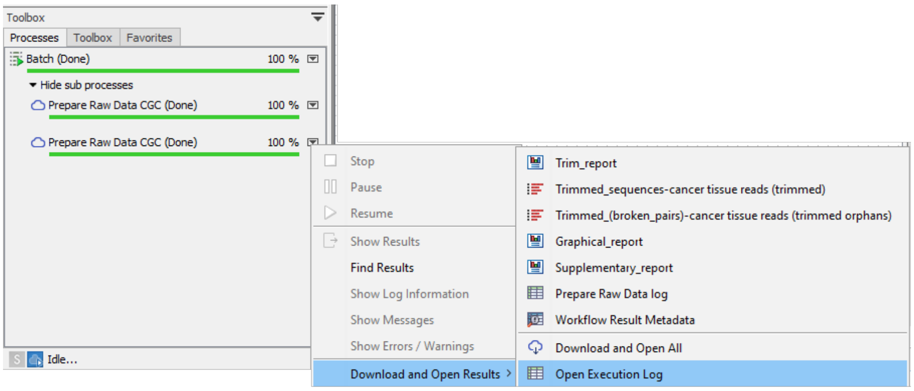


Figure 7.1: *Results and log files for a given analysis run on the cloud can be accessed from the Processes tab in the Workbench.*

To clear processes from the list in the Processes tab, right-click on an empty area and select the option **Remove Finished Processes**.

After processes have been cleared, the same list of results, and access to the Execution Log is available by double clicking on a `workflow-result.json` file under the Remote Files tab (figure 7.2).

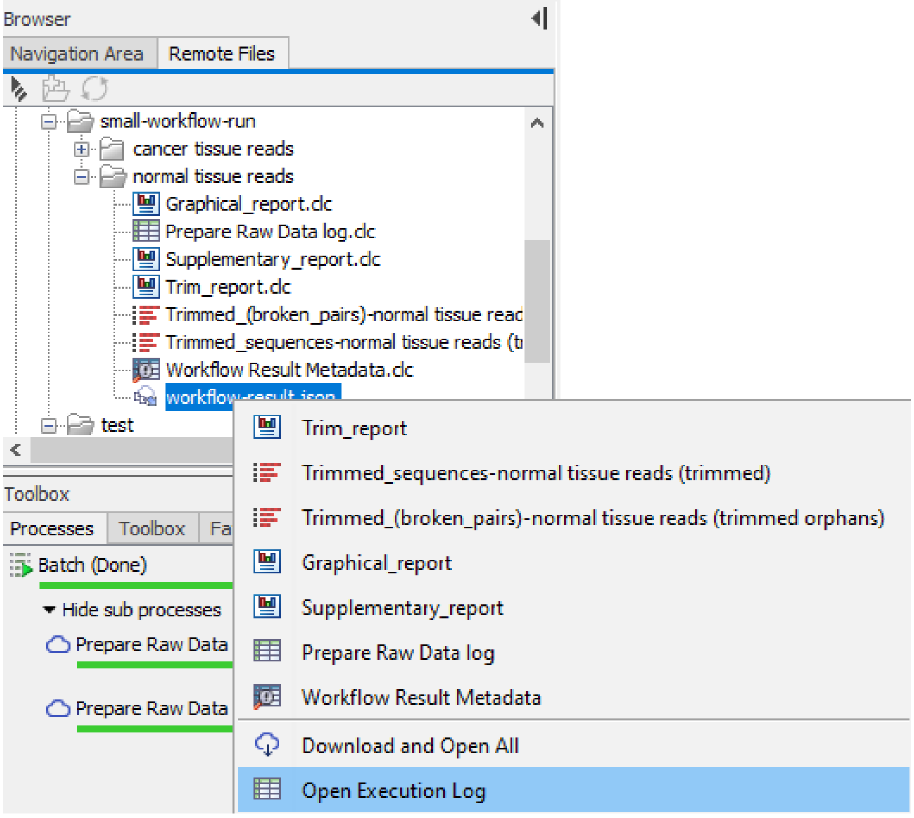
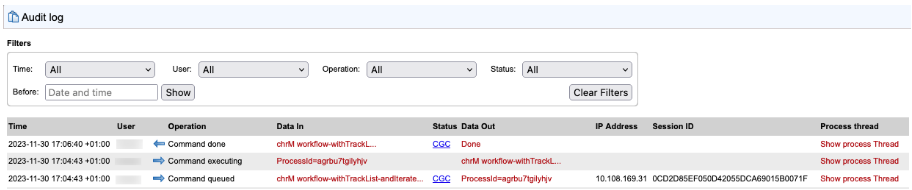


Figure 7.2: *Double-clicking on a workflow-result.json file in the Remote Files tab in the Workbench provides access to results and log files for a given analysis run on the cloud.*

7.2 Using a CLC Server to monitor and review cloud jobs

Information about queued and running jobs submitted via the *CLC Server* is available under in the Queue tab, under the Management tab of the *CLC Server* web administrative interface. Information about queued, running and completed processes submitted via the *CLC Server* is available from the Audit log, also under the Management tab of the web administrative interface. Cloud jobs have a have a **CGC** link in the Status column (figure 7.3).


Audit log

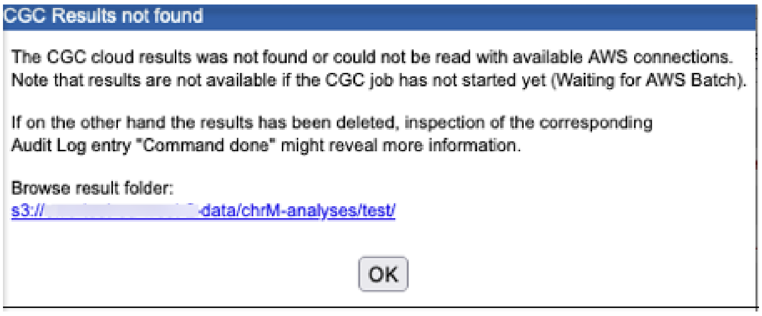
Filters

Time: All
User: All
Operation: All
Status: All
Before: Date and time

Time	User	Operation	Data In	Status	Data Out	IP Address	Session ID	Process thread
2023-11-30 17:06:40 +01:00	[redacted]	Command done	chrM workflow-withTrackL...	CGC	Done			Show process Thread
2023-11-30 17:04:43 +01:00	[redacted]	Command executing	ProcessId=agrbu7tgilyhjv		chrM workflow-withTrackL...			Show process Thread
2023-11-30 17:04:43 +01:00	[redacted]	Command queued	chrM workflow-withTrackList-andIterate...	CGC	ProcessId=agrbu7tgilyhjv	10.108.169.31	0CD2D85EF050D42055DCA69015B0071F	Show process Thread

Figure 7.3: *An audit log entry for a completed job. The CGC links in the Status column indicate this was a job sent to the cloud for execution.*

When a cloud job is in progress, clicking on the **CGC** link beside a job in the Queue or in the Status column of the audit log opens a dialog. Before the job starts running, the dialog gives some basic information and a link to the results folder (figure 7.4). While a job is being executed, the AWS log viewer will be opened, allowing progress of the job to be monitored via the CloudWatch log. Clicking on the **Reload log** button reloads the log and scrolls to latest entry. Search functionality is available at the bottom left (figure 7.5).



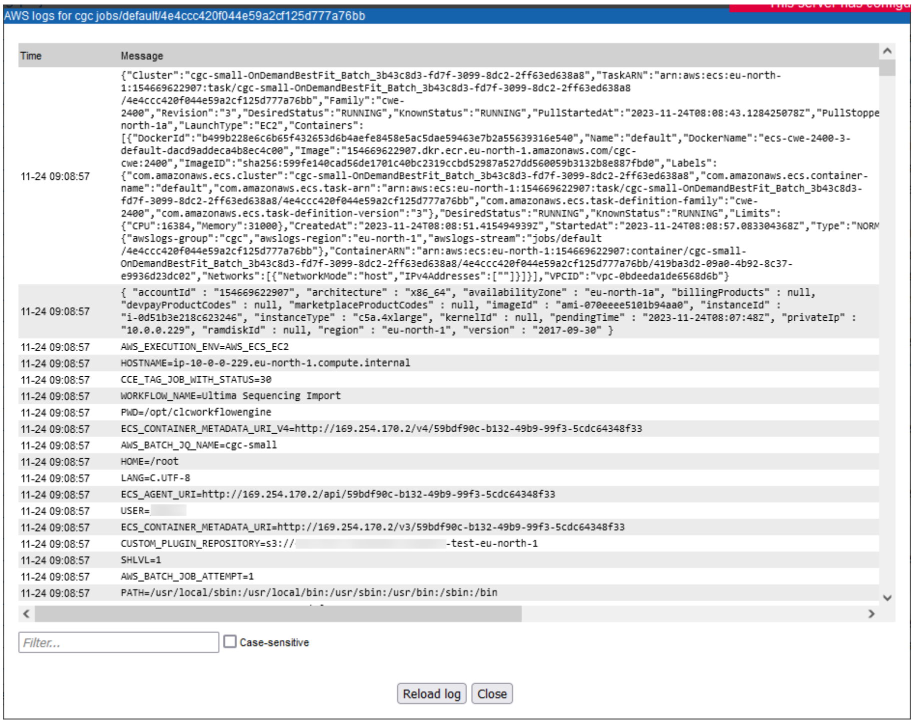
CGC Results not found

The CGC cloud results was not found or could not be read with available AWS connections.
Note that results are not available if the CGC job has not started yet (Waiting for AWS Batch).

If on the other hand the results has been deleted, inspection of the corresponding
Audit Log entry "Command done" might reveal more information.

Browse result folder:
s3://[redacted]-data/chrM-analyses/test/

Figure 7.4: *Clicking on the CGC link for a queued job opens a dialog with basic information and a link to the results folder.*



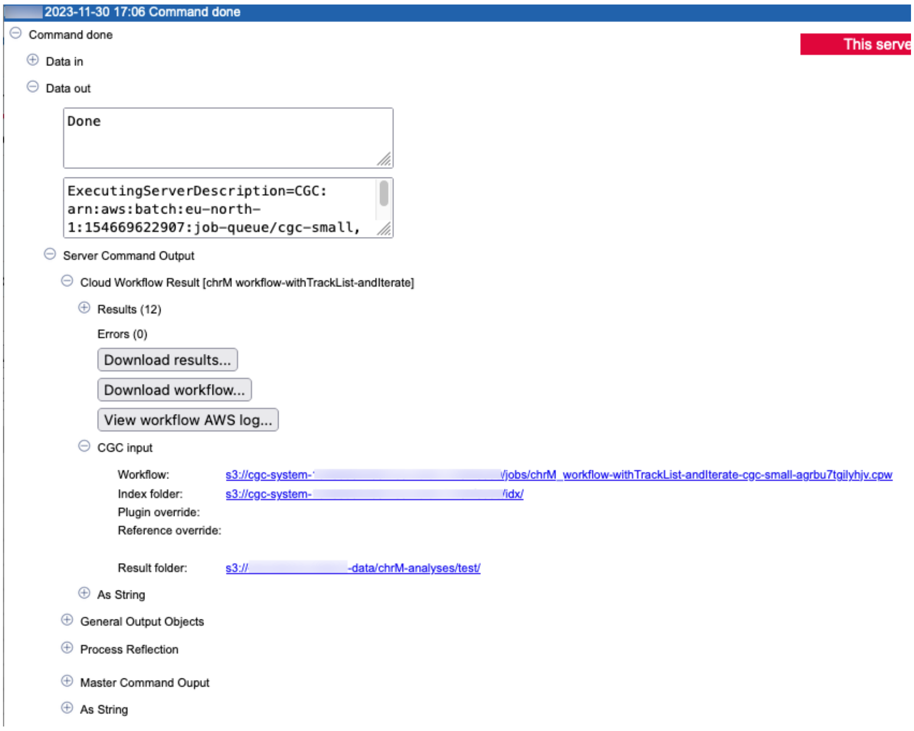
AWS logs for cgc jobs/default/4e4ccc420f044e59a2cf125d777a76bb

```
Time             Message
                 {"Cluster":"cgc-small-OnDemandBestFit_Batch_3b43c8d3-fd7f-3099-8dc2-2ff63ed638a8","TaskARN":"arn:aws:ecs:eu-north-
                 1:154669622907:task/cgc-small-OnDemandBestFit_Batch_3b43c8d3-fd7f-3099-8dc2-2ff63ed638a8
                 /4e4ccc420f044e59a2cf125d777a76bb","Family":"cwe-
                 2400","Revision":"3","DesiredStatus":"RUNNING","KnownStatus":"RUNNING","PullStartedAt":"2023-11-24T08:08:43.128425078Z","PullStoppe
                 north-1a","LaunchType":"EC2","Containers":
                 [{"DockerId":"b499b228e6c6b65f432653d6b4aefe8458e5ac5dae59463e7b2a55639316e540","Name":"default","DockerName":"ecs-cwe-2400-3-
                 default-dacd9addeca4b8ec4c00","Image":"154669622907.dkr.ecr.eu-north-1.amazonaws.com/cgc-
                 cwe:2400","ImageID":"sha256:599fe140cad56de1701c40bc2319ccbd52987a527dd560059b3132b8e887fbd0","Labels":
11-24 09:08:57   {"com.amazonaws.ecs.cluster":"cgc-small-OnDemandBestFit_Batch_3b43c8d3-fd7f-3099-8dc2-2ff63ed638a8","com.amazonaws.ecs.container-
                 name":"default","com.amazonaws.ecs.task-arn":"arn:aws:ecs:eu-north-1:154669622907:task/cgc-small-OnDemandBestFit_Batch_3b43c8d3-
                 fd7f-3099-8dc2-2ff63ed638a8/4e4ccc420f044e59a2cf125d777a76bb","com.amazonaws.ecs.task-definition-family":"cwe-
                 2400","com.amazonaws.ecs.task-definition-version":"3"},"DesiredStatus":"RUNNING","KnownStatus":"RUNNING","Limits":
                 {"CPU":16384,"Memory":31000},"CreatedAt":"2023-11-24T08:08:51.415494939Z","StartedAt":"2023-11-24T08:08:57.083304368Z","Type":"NORM
                 {"awslogs-group":"cgc","awslogs-region":"eu-north-1","awslogs-stream":"jobs/default
                 /4e4ccc420f044e59a2cf125d777a76bb"},"ContainerARN":"arn:aws:ecs:eu-north-1:154669622907:container/cgc-small-
                 OnDemandBestFit_Batch_3b43c8d3-fd7f-3099-8dc2-2ff63ed638a8/4e4ccc420f044e59a2cf125d777a76bb/419ba3d2-09a0-4b92-8c37-
                 e9936d23dc02","Networks":[{"NetworkMode":"host","IPv4Addresses":[""]}]}],"VPCID":"vpc-0bdeeda1de6568d6b"}
                 { "accountId" : "154669622907", "architecture" : "x86_64", "availabilityZone" : "eu-north-1a", "billingProducts" : null,
11-24 09:08:57   "devpayProductCodes" : null, "marketplaceProductCodes" : null, "imageId" : "ami-070eeee5101b94aa0", "instanceId" :
                 "i-0d51b3e218c623246", "instanceType" : "c5a.4xlarge", "kernelId" : null, "pendingTime" : "2023-11-24T08:07:48Z", "privateIp" :
                 "10.0.0.229", "ramdiskId" : null, "region" : "eu-north-1", "version" : "2017-09-30" }
11-24 09:08:57   AWS_EXECUTION_ENV=AWS_ECS_EC2
11-24 09:08:57   HOSTNAME=ip-10-0-0-229.eu-north-1.compute.internal
11-24 09:08:57   CCE_TAG_JOB_WITH_STATUS=30
11-24 09:08:57   WORKFLOW_NAME=Ultima Sequencing Import
11-24 09:08:57   PWD=/opt/clcworkflowengine
11-24 09:08:57   ECS_CONTAINER_METADATA_URI_V4=http://169.254.170.2/v4/59bdf90c-b132-49b9-99f3-5cdc64348f33
11-24 09:08:57   AWS_BATCH_JQ_NAME=cgc-small
11-24 09:08:57   HOME=/root
11-24 09:08:57   LANG=C.UTF-8
11-24 09:08:57   ECS_AGENT_URI=http://169.254.170.2/api/59bdf90c-b132-49b9-99f3-5cdc64348f33
11-24 09:08:57   USER=[redacted]
11-24 09:08:57   ECS_CONTAINER_METADATA_URI=http://169.254.170.2/v3/59bdf90c-b132-49b9-99f3-5cdc64348f33
11-24 09:08:57   CUSTOM_PLUGIN_REPOSITORY=s3://[redacted]-test-eu-north-1
11-24 09:08:57   SHLVL=1
11-24 09:08:57   AWS_BATCH_JOB_ATTEMPT=1
11-24 09:08:57   PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
```

Figure 7.5: *Clicking on the CGC link for a running job opens the AWS log viewer. Contents can be searched by typing terms into the field at the bottom left. Clicking on the Reload log button reloads the log and scrolls to the latest entry.*

When a job is complete, the word Done will be present in an audit log entry for that job. Clicking on a link in red text in that row opens a window containing detailed information about the job. Drill down through the Server Command Output to get to buttons for downloading the AWS log, i.e. the Cloud Watch log stream for that job, as well as buttons for downloading results or downloading a copy of the workflow that was run.

Clicking on the **CGC** link in the Status column of completed jobs gives direct access to the Cloud Workflow Result section of the audit log entry, as does navigating to a workflow-result.json file in the **Browse S3 locations** area under the Element info tab of the *CLC Server* web interface and clicking on the **View workflow result...** button (figure 7.7).



[redacted] 2023-11-30 17:06 Command done

Command done

Data in

Data out

```
Done
```

```
ExecutingServerDescription=CGC:
arn:aws:batch:eu-north-
1:154669622907:job-queue/cgc-small,
```

Server Command Output

Cloud Workflow Result [chrM workflow-withTrackList-andIterate]

Results (12)

Errors (0)

CGC input

Workflow: s3://cgc-system-[redacted]/jobs/chrM_workflow-withTrackList-andIterate-cgc-small-agrbu7tgilyhjv.cpw
Index folder: s3://cgc-system-[redacted]/idx/
Plugin override:
Reference override:

Result folder: s3://[redacted]-data/chrM-analyses/test/

As String

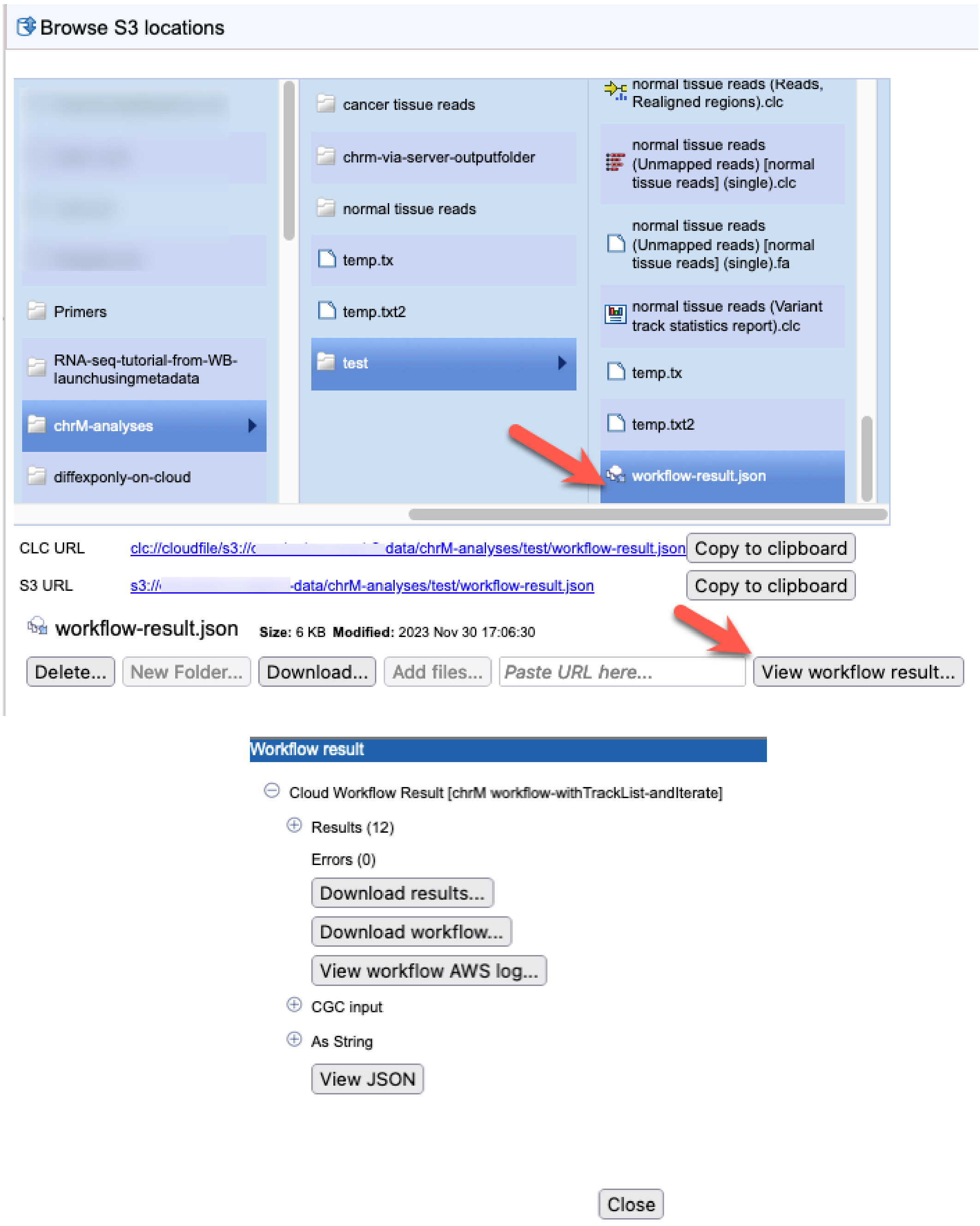
General Output Objects

Process Reflection

Master Command Ouput

As String

Figure 7.6: *In the Command done window, buttons are provided under Server Command Output that can be used to view the AWS log, i.e. the CloudWatch log stream, as well as for downloading results or the workflow run. Links to input data and the results folder are also available.*



CLC URL clc://cloudfile/s3://[redacted]data/chrM-analyses/test/workflow-result.json

S3 URL s3://[redacted]-data/chrM-analyses/test/workflow-result.json

workflow-result.json **Size:** 6 KB **Modified:** 2023 Nov 30 17:06:30

Cloud Workflow Result [chrM workflow-withTrackList-andIterate]

Figure 7.7: *Buttons to download workflow results and the AWS log, i.e. the CloudWatch log stream, can be revealed by navigating to a workflow-result.json file in the Browse S3 locations area and clicking on the "View workflow result..." button.*

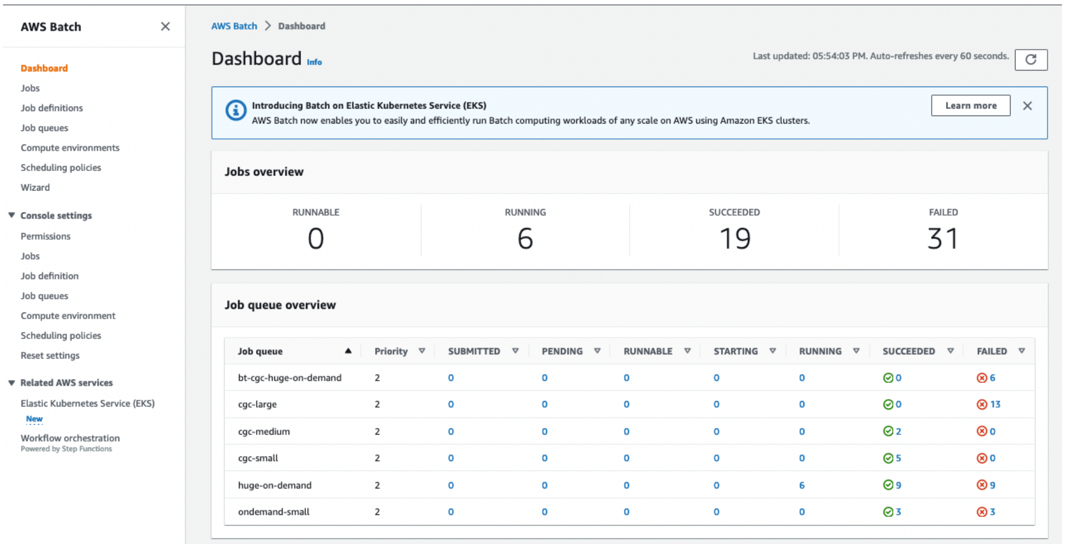
By default, the Audit Log and Queue tabs in the *CLC Server* web interface are only available to administrative users. Access can be granted to users of other groups, as described at

`https://resources.qiagenbioinformatics.com/manuals/clcserver/current/admin/index.php?manual=Controlling_access_server_server_tasks_external_data.html`.

7.3 Using the AWS Console to monitor and review cloud jobs

CLC jobs run on AWS are monitored using CloudWatch. The log group **cgc** contains log streams for each job. The AWS_BATCH_JOB_ID field, near the top of the log information, links the log stream to the job ID.

You can navigate to the job log in CloudWatch from the AWS Batch page in the AWS Console. The Job queue overview in the Batch dashboard contains links to information about jobs in each queue, according to their status (e.g. Running, Succeeded, Failed). Click on one of these links to see a list of the jobs in that category for that queue (figure 7.8). From there, click on the link for an individual job and then click on the logstream link, in the Job information section. This takes you to the information about that job in CloudWatch.



Last updated: 05:54:03 PM. Auto-refreshes every 60 seconds.

Introducing Batch on Elastic Kubernetes Service (EKS)
AWS Batch now enables you to easily and efficiently run Batch computing workloads of any scale on AWS using Amazon EKS clusters.

Jobs overview

RUNNABLE	RUNNING	SUCCEEDED	FAILED
0	6	19	31

Job queue overview

Job queue	Priority	SUBMITTED	PENDING	RUNNABLE	STARTING	RUNNING	SUCCEEDED	FAILED
bt-cgc-huge-on-demand	2	0	0	0	0	0	✓ 0	⊗ 6
cgc-large	2	0	0	0	0	0	✓ 0	⊗ 13
cgc-medium	2	0	0	0	0	0	✓ 2	⊗ 0
cgc-small	2	0	0	0	0	0	✓ 5	⊗ 0
huge-on-demand	2	0	0	0	0	6	✓ 9	⊗ 9
ondemand-small	2	0	0	0	0	0	✓ 3	⊗ 3

Figure 7.8: *Information about jobs run on a CLC Genomics Cloud setup can be found using AWS functionality, such as the AWS Batch overview, shown here. Clicking on the links on this page allow you to drill down to details about individual jobs.*

Chapter 8

Appendix

8.1 Standard AWS Batch queues for CLC Genomics Cloud

The three AWS Batch queues defined in the CloudFormation template at:

`https://qiagen-clc-genomics-cloud-formation.s3.eu-central-1.amazonaws.com/cgc-standard.json`

have the following settings:

cgc-small

Setting	**Value**
EC2 instance type	c5a.4xlarge
vCPUs per job	16
Max vCPUs	160
Total available memory per job	31000 MB
Java available memory per job	15500 MB
Disk space	100 GB

cgc-medium

Setting	**Value**
EC2 instance type	c5a.8xlarge
vCPUs per job	32
Max vCPUs	320
Total available memory per job	63000 MB
Java available memory per job	31000 MB
Disk space	300 GB

cgc-large

Setting	Value
EC2 instance type	c5a.16xlarge
vCPUs per job	64
Max vCPUs	640
Total available memory per job	127000 MB
Java available memory per job	50000 MB
Disk space	500 GB

8.2 AWS Batch queues, compute environments, AMIs and images

AWS Batch queues, compute environments and AMIs

AWS Batch queues created using the CloudFormation templates provided by QIAGEN are each mapped to an individual compute environment. Each compute environment is configured to use the latest Amazon ECS-optimized AMI available at the time it is established. Thus, in a CLC Genomics Cloud setup, all jobs sent to a given AWS Batch queue will use the same AMI version.

AWS Batch does not upgrade the AMIs in existing compute environments. Thus, compute environments established at different times may contain different AMI versions. An example where this could happen in a CLC Genomics Cloud context would be where a new AWS Batch queue is created some time after the initial infrastructure was set up, and there had been an AMI update released in the meantime.

We generally do not expect performance issues using supported, older AMI versions. From a security perspective, AWS Batch compute environments established using the CloudFormation templates provided by QIAGEN are placed in a security group that rejects incoming traffic that was not initiated from the EC2 host.

The AMI used for a job

The image ID for the AMI used for a job is recorded near the top of the CloudWatch log (aka Execution Log). Information about that public image, such as the Amazon Linux AMI version and the deprecation time for that AMI can be found via the AWS Console:

Console | EC2 | Images | AMIs

See the changelog at AWS for details of each AMI release: `https://github.com/aws/amazon-ecs-ami/blob/main/CHANGELOG.md`

Accessing a newer AMI

To update the AMI being used, a new compute environment is needed. An easy way to achieve this is to create a new AWS Batch queue, as described in section 2.2. Queues that are no longer needed can be deleted, as described in section 2.4.

CLC software and plugin versions used for a job

CLC software version used: Execution of CLC jobs using AWS Batch involves pulling the Docker image from AWS ECR that contains the CLC Workflow Engine version corresponding to the version of the CLC product used to submit the cloud job. For example:

- A job submitted directly from CLC Genomics Workbench 23.0.4 will pull the image containing CLC Workflow Engine 23.0.4.
- A job submitted via CLC Genomics Server 23.0.5 will pull the image containing CLC Workflow Engine 23.0.5. This is the case irrespective of the client software version used to submit the job to the CLC Server.

The CLC Workflow Engine is released in conjunction with the CLC Main Workbench, CLC Genomics Workbench, CLC Genomics Server, and CLC Server Command Line Tools. Thus, there is a CLC Workflow Engine version corresponding to every released CLC Workbench and CLC Genomics Server version capable of submitting jobs to AWS Batch.

Plugin versions used: Each plugin contributing tools needed for the analysis is installed in the CLC Workflow Engine before the analysis is run. The version of each plugin installed on the CLC Workflow Engine is the same as the version installed on the CLC product used to submit the cloud job. E.g. When jobs are submitted via a CLC Genomics Server, it is the version of the plugin on the CLC Server that will be installed on the CLC Workflow Engine.

8.3 System file retention policies

By default, system files in the CGC system S3 bucket are deleted after 30 days, and read mapper indexes are deleted 14 days after creation. These defaults can be changed by editing the expire-jobs and expire-read-mapper-index-files lifecycle rules associated with the CGC system bucket.

Note: If the same reference genome or genomes will generally be used for mappings, extending the period read mapper indexes are retained may be of interest. Deleting the files every 14 days could result in the indexes being rebuilt more often than necessary.

8.4 Making custom plugins available for cloud analyses

Tools[1] delivered by custom CLC plugins are supported for cloud analyses when:

- Server plugins are placed in a plugin repository that has the expected file organization (described below).
- The server plugins have the name `<workbench_plugin_id>.cpa`.
- The location of the custom plugin repository is configured when setting up the standard CLC Genomics Cloud infrastructure on AWS, as described at section 2.1.

When the above requirements have been met, analyses using tools delivered by a custom plugin can be run on the cloud. The corresponding plugin in the custom repository is used for the cloud-based execution.

---

[1] Only tools that can be used within workflows can be submitted to run on a CLC Genomics Cloud. Tools that are not workflow-enabled cannot be run on the cloud.

Custom plugin repository structure

The custom plugin repository must have the structure and contents defined below. The term `<workbench_plugin_id>` refers to the value of that field in the plugin.properties file.

- A top level folder per plugin. The folder name must be `<workbench_plugin_id>`.
- A subfolder named `server`.
- Within the `server` folder, a subfolder with the name of the *version of the plugin*. e.g. "23.0" or "22.0.2".

  This name must be the version of the plugin. This may be different to the version or your CLC Workbench or CLC Server.
- The plugin .cpa file and any other relevant files (e.g. plugin.properties) must be placed in this folder.

  The plugin .cpa file must be named as `<workbench_plugin_id>.cpa`.

For example, a server plugin with the `<workbench_plugin_id` value `example_plugin` and version `23.0`, the following structure in the plugin repository would be expected:

```
example_plugin/
                    server/
                             23.0/
                                   example_plugin.cpa
                                   plugin.properties
```

When a job is launched to run on the cloud, the exact version of the plugin in the CLC Workbench or CLC Server must be available in the repository. If it is not, the job will fail.