Collibra

Collibra Cloud Self-Hosted

# Installation and upgrade guide

Version 2024.01

Collibra

# Collibra Cloud Self-Hosted - Installation and upgrade

Release date: November 19, 2023

Revision date: January 02, 2024

You can find the most up-to-date technical documentation on our Documentation Center at

https://productresources.collibra.com/docs/collibra/latest/Content/Installation/CCSH/to_ccsh-install-upgrade.htm

# Contents

# Collibra Cloud Self-Hosted getting started

Thanks so much for your continued support of our vision to do more with trusted data. This guide helps explain updates we've made to our solution so you can quickly orient yourself. This is not intended to be a comprehensive guide but should serve as a good starting point for experiencing everything that Collibra Cloud Self-Hosted has to offer.

## Documentation toggle

Please note that for this release, the release notes contain a "CCSH" toggle. If you turn this toggle on, it will filter out irrelevant content that does not apply to the CCSH solution. We have done our best to ensure we've captured the right information for you but please note that there are some things that may still have varying impacts for each platform. If you find a documentation issue that we didn't catch, don't hesitate to contact us to fix it!

## How to upgrade from 5.9.1 (or later) to 2023.11?

Upgrading to the newest version of Collibra Cloud Self-Hosted is intended to be seamless. Follow our documentation to see how little has changed in the upgrade process!

> Note If you are using Collibra Data Quality & Observability on-premises, we recommend you upgrade to at least the 2023.09 release. There are several key fixes in that version which make your upgrade much more seamless.

## New features in 2023.11

It wouldn't be a new release if it didn't have new features! As part of our effort to deliver more for our CCSH customers, we've added new features for you. Expect to see more from us very shortly as we continue to improve this product.

- Data Marketplace
    - Your new CCSH deployment also includes Data Marketplace, the best place to find, request and check out data for use! Read all about it in the product documentation.

      > Note  Make sure to disable Data Discovery in the Data Marketplace settings so you see the right cards on the Data Marketplace landing page.

- Homepage
    - This version of Collibra includes Homepage, a new feature, designed to present for your users the most relevant information immediately! Read up on the new features here! Please note that some features of Homepage will not be available.

# Major changes from 5.9.x to 2023.11

2023.11 brings a log of code updates across the board from enhanced security to new features to improved stability. See a list of the biggest changes here:

- **Monitoring service**: We've replace the Monitoring service with a more robust, more standard monitoring mechanism. The Monitoring service was constrained and highly prescriptive. With our new logging and monitoring setup, you can ingest data into your own alerting and visualization systems as you please. For more information about diagnostic files, go to the Platform Configuration section.
- **API v1 Endpoints**: We're in the process of upgrading our API services to v2. As a result, some of the supported APIs have changed with CCSH. Please read through the documentation to check for any impacts.
- **Asset Import UI**: The Asset Import UI has changed. In this release, we've preserved the older UI but flagged it off. If you require it, our Support teams can assist you with re-enabling. However, we recommend moving to the new UI as the 2024.02 GA release of Collibra Cloud Self-Hosted will finally remove the old UI entirely.

- **Workflow APIs**: Some Workflow APIs may be deprecated as a result of the move to Workflow Designer. We have a tool available to help you test if your workflows are impacted but if you find something not presently documented, please don't hesitate to contact the support team.
- **Homepage**: CCSH customers do not have access to the Most Viewed Assets and Recommended Assets sections of the product as both cloud features rely on a piece of software we are unable to ship externally. We'll review possible ways to make this available in the future for CCSH customers!

# Contact the Public Beta team

If you find anything not to your liking with the Collibra Cloud Self-Hosted Beta release, please don't hesitate to reach out to Collibra! First and foremost, your point of contact with Collibra will always be your Customer Support Manager or Account Executive. They can always reach the Product team with any feedback!

However, we'd also love to hear feedback via our ideation platform. We respond to all feedback and we are sincerely grateful for any help you can provide in improving both our General Availability offering as well as our future for CCSH customers.

# Feature availability comparison

Collibra Cloud Self-Hosted (CCSH) is a Collibra solution that allows you to install Collibra Data Intelligence Cloud on an infrastructure of your choice. This solution allows you to continue using Collibra in an on-premises environment. Compared to our previous on-premises offering, Collibra Data Governance Center 5.9.1 and older, new features that were previously only available in our cloud offering, are added, though there are still differences between these offerings.

The following table is an overview of the feature availability in Collibra Data Intelligence Cloud, Collibra Cloud Self-Hosted (CCSH), and Collibra Data Governance Center.

| Product | Collibra Data Intelligence Cloud for commercial customers (AWS and GCP) | Collibra Cloud for Government | Collibra Cloud Self-Hosted | Collibra DGC 5.9/5.9.1 |
|---|---|---|---|---|
| Auto Classification | ✔ Yes | ✖ No | ✖ No | ✖ No |
| Data Governance | ✔ Yes | ✔ Yes | ✔ Yes | ✔ Yes |
| Data Catalog | ✔ Yes | ✔ Yes | ✔ Yes | ✔ Yes |
| Data Privacy | ✔ Yes | ✔ Yes | ✔ Yes | ✔ Yes |
| Technical Lineage | ✔ Yes | ✔ Yes | ✖ No | ✖ No |
| Data Marketplace | ✔ Yes | ✖ No (*) | ✔ Yes | ✖ No |

| Product | Collibra Data Intelligence Cloud for commercial customers (AWS and GCP) | Collibra Cloud for Government | Collibra Cloud Self-Hosted | Collibra DGC 5.9/5.9.1 |
|---|---|---|---|---|
| Collibra Data Quality & Observability | ✓ Yes | ✓ Yes | ✓ Yes | ✕ No |
| Edge | ✓ Yes | ✓ Yes | ✕ No | ✕ No |
| Homepage | ✓ Yes | ✕ No (*) | ✓ Yes (**) | ✕ No |
| Insights | ✓ Yes | ✓ Yes | ✕ No | ✕ No |
| Collibra Protect | ✓ Yes | ✕ No | ✕ No | ✕ No |
| Usage Analytics | ✓ Yes | Planned for first half of 2024 | ✕ No | ✕ No |
| Workflow Designer | ✓ Yes | ✕ No | ✕ No | ✕ No |

(*) Available soon!

(**) Homepage will not be enabled by default for new and upgraded environments. Follow the instructions of the Enable Homepage section to enable it once you are on Collibra Cloud Self-Hosted 2023.11.

Chapter 3

# Install Collibra Cloud Self-Hosted

This section describes how to install Collibra Cloud Self-Hosted.

# Collibra Cloud Self-Hosted installation requirements

Before you start the installation, you need all of the following information to ensure an easy, successful installation process. This section only focuses on the requirements of the core platform and does not take into account the connections to the data sources to ingest data.

## System requirements

### Supported operating systems

You can install the Collibra Cloud Self-Hosted solution only on the following operating systems:

- Red Hat Enterprise Linux/CentOS 7
- Red Hat Enterprise Linux/Rocky Linux 8
- Red Hat Enterprise Linux/Rocky Linux 9
- Windows Server 2016 2019, and 2022

> Important
> - Only the x86_64 architecture is supported.
> - On Linux, you can use root and standard users for the installation.
> - On Windows, you need administrative rights for the installation.

## Hardware requirements

| Number of concurrent users | Number of assets | Recommended requirements for DGC service | Recommended requirements for Repository |
|---|---|---|---|
| 1 - 50 | < 1 million | 4 CPUs / 16 GB memory | 2 CPUs / 16 GB memory |
| 50 - 100 | 1 - 5 million | 4 CPUs / 16 GB memory | 4 CPUs / 16 GB memory |

| Number of concurrent users | Number of assets | Recommended requirements for DGC service | Recommended requirements for Repository |
|---|---|---|---|
| 50 - 200 | 5 - 25 million | 8 CPUs / 32 GB memory | 8 CPUs / 32 GB memory |
| 200 - 500 | 25 - 50 million | 16 CPUs / 64 GB memory | 16 CPUs / 64 GB memory |
| > 500 | > 50 million | 32 CPUs / 128 GB memory | 32 CPUs / 128 GB memory |

## Network requirements

Collibra Cloud Self-Hosted uses the following ports for the following services.

| Port | Default value | Purpose |
|---|---|---|
| Agent application | 4401 | TCP port used by Collibra Console to manage the services in a Collibra environment. |
| Console application | 4402 | TCP port to access your Collibra Console via your web browser. |
| Console database | 4420 | TCP port to access the database of Collibra Console. |
| Collibra Data Quality & Observability | 80 | TCP port to ingest Collibra DQ metadata over REST API. |
| DGC service, including Assessments, Usage Analytics, Privacy, Protect | 4400 | TCP port to access your Collibra environment via your web browser. |
| DGC shutdown port | 4430 | TCP port through which you can stop the DGC service. |
| Insights Data Access | 443 | TCP port to access Insights Data Access. |

| Port | Default value | Purpose |
|---|---|---|
| Jobserver database | 4414 | TCP port to access the Jobserver database. |
| Jobserver monitoring port | 4424 | Port used by the Monitoring service to monitor the Jobserver service. |
| Jobserver service | 4404 | TCP port to access the Jobserver service. |
| Jobserver Spark monitoring port | 4434 | Port used by the Monitoring service to monitor the Spark service. |
| Repository service | 4403 | TCP port to access the repository service. It is used only by the DGC service and the Collibra agent. |
| Search HTTP port | 4421 | TCP port to access the Search service. |
| Search Transport port | 4422 | TCP port used by the DGC service to communicate with the Search service. |

# Prepare the servers for installation

## General preparation

- We recommend that you have 4 dedicated nodes: one for the Data Governance Center service and Search service, one for Collibra Console, one for the Repository service, and one for Jobserver if applicable. Make sure that you have a fast network between the nodes.
- You have downloaded the Collibra Cloud Self-Hosted installer on all nodes.

## PostgreSQL 14.9 on Linux

If you are installing Collibra Cloud Self-Hosted on Linux, you must install PostgreSQL 14, on any server that runs one of the following services:

- Repository
- Jobserver

- Collibra Console

The current latest version on Linux is 14.9.

> Note PostgreSQL 14.9 is included in the Windows installer and will be automatically installed during the CCSH installation.

## Install PostgreSQL 14.9 with Internet connection

1. Install PostgreSQL 14.9 with the following commands.

   > Important Run the commands as root.

   ```
   #Clean the YUM cache and update existing packages for your
   current Linux repository. Note that this makes system
   changes.

   yum clean all && yum update -y

   #Prepare the PostgreSQL repository and packages:
   yum -y install
   https://download.postgresql.org/pub/repos/yum/reporpms/EL-
   $(rpm -E %{rhel})-x86_64/pgdg-redhat-repo-latest.noarch.rpm

   #Update the packages in the repository:
   yum -y update

   #Install the PostgreSQL 14.9 packages:
   yum -y install postgresql14 postgresql14-server
   postgresql14-contrib
   ```

2. Update the file **/usr/lib/tmpfiles.d/postgresql-14.conf** to set the correct permissions for some PostgreSQL 14.9 folders. Open the file for editing, for example with vim or nano and update the line `d /run/postgresql 0755 postgres postgres -` to:

   ```
   d /run/postgresql 2777 postgres postgres - -
   ```

> Important Do not use the `chmod` command on any directories or files, edit this configuration file instead.

3. Reboot the server.

> Tip The default PostgreSQL 14.9 path on RHEL/Rocky/CentOS is **/usr/pgsql-14**.

## Install PostgreSQL 14.9 without Internet connection

### Download the necessary PostgreSQL 14.9 packages

On a server with Internet access:

1. Add the PostgreSQL repository:

```
sudo yum -y install
https://download.postgresql.org/pub/repos/yum/reporpms/EL-
$(rpm -E %{rhel})-x86_64/pgdg-redhat-repo-latest.noarch.rpm
```

2. Update the package list:

```
sudo yum -y update
```

3. Install the yum-utils:

```
sudo yum install -y yum-utils
```

4. Get the web links for the necessary PostgreSQL packages and their dependencies:

```
yumdownloader --urls -y postgresql14 postgresql14-contrib
postgresql14-server
```

5. Download packages and their dependencies:

```
yumdownloader --resolve -y postgresql14 postgresql14-
contrib postgresql14-server
```

> Important The list and names of the packages depend on your distro version. The following packages must always be included:
> - lz4-1.9.3-5.el9.x86_64.rpm
> - postgresql14-14.9-2PGDG.rhel9.x86_64.rpm
> - postgresql14-contrib-14.9-2PGDG.rhel9.x86_64.rpm
> - postgresql14-libs-14.9-2PGDG.rhel9.x86_64.rpm
> - postgresql14-server-14.9-2PGDG.rhel9.x86_64.rpm

6. Copy the downloaded packages to the server on which you want to install PostgreSQL 14.9.

## Install PostgreSQL 14.9

1. From the directory that contains the packages, run the following command to install the PostgreSQL packages:

```
sudo yum localinstall -y *.rpm
```

2. Update the file **/usr/lib/tmpfiles.d/postgresql-14.conf** to set the correct permissions for some PostgreSQL 14.9 folders. Open the file for editing, for example with vim or nano and update the line `d /run/postgresql 0755 postgres postgres -` to:

```
d /run/postgresql 2777 postgres postgres - -
```

> Important Do not use the `chmod` command on any directories or files, edit this configuration file instead.

3. Reboot the server.

## Operating system configuration for Search service

The node that will run the Search service, must pass the following bootstrap checks:

- File descriptor
- Maximum number of threads check
- Maximum file size
- Maximum size virtual memory check
- Maximum map count check

| Type | Check description | Minimum value | Applies for installation type | Setting name |
|---|---|---|---|---|
| User limit | Maximum number of open file descriptors | 65536 | ▪ Without root permissions<br>▪ With root permissions, using System V init daemon | nofile |
| | Maximum number of open threads/processes | 4096 | | nproc |
| | Maximum file size | unlimited | | fsize |
| Kernel parameter | Maximum virtual memory areas | 262144 | ▪ All | vm.max_map_count |

For more information on these settings, see the Troubleshooting section.

# Install Collibra Data Intelligence Cloud on Collibra Cloud Self-Hosted

This section describes the installation of Collibra Cloud Self-Hosted.

# Steps

> Tip
> There is no graphical user interface for the installer. The full installation procedure is executed via the command line.
>
> For each question, the default selection is always suggested between square brackets. Press `Enter` to accept the default selection. If there is a Yes or No question, the upper-case character is the default selection, for example, in [Y/n] the default selection is Yes.

## Linux installation as root

> Note If you want to configure the init daemon on Linux systems, you have to execute an unattended installation. For more information, go to the unattended installation configuration parameters.

1. Run the installer:
   Linux as user with sudo rights: `sudo ./dgc-linux-2024.01.0.sh`
   Linux as root user: `./dgc-linux-2024.01.0.sh`
2. Enter the **Installation directory** and press `Enter`.
   The default location on Linux is **/opt/collibra**.
3. Enter the **Data directory** and press `Enter`.
   The default location on Linux is **/opt/collibra_data**.
4. Press `Enter` to each of the presented components that you want to install.
   If you don't want to install a specific component, press `n` followed by `Enter`.

   > Note We recommend that you have 4 dedicated nodes: one for the Data Governance Center service and Search service, one for Collibra Console, one for the Repository service, and one for Jobserver if applicable. Make sure that you have a fast network between the nodes.
   > Also make sure that you use the same installer version on all nodes. You can find the installer version of your environment at the bottom of the sign-in window of Collibra Console, for example 2024.01.0

5. Press `Enter` to confirm your selection.
6. If you have selected Repository, Jobserver and/or Collibra Console, enter the location where PostgreSQL 14.9 is installed.

   > Tip The default PostgreSQL 14.9 path on RHEL/Rocky/CentOS is **/usr/pgsql-14**.

7. Enter the necessary configuration for each of the selected services.
   » After the last configuration, the installation of the services automatically starts.

## Linux installation as standard user

1. Run the installer: `./dgc-linux-2024.01.0.sh`
2. Enter the **Installation directory** and press `Enter`.
   The default location on Linux as standard user is **~/collibra**.
3. Enter the **Data directory** and press `Enter`.
   The default location on Linux as standard user is **~/collibra_data**.
4. Press `Enter` to each of the presented components that you want to install.
   If you don't want to install a specific component, press `n` followed by `Enter`.

   > Note We recommend that you have 4 dedicated nodes: one for the Data Governance Center service and Search service, one for Collibra Console, one for the Repository service, and one for Jobserver if applicable. Make sure that you have a fast network between the nodes.
   > Also make sure that you use the same installer version on all nodes. You can find the installer version of your environment at the bottom of the sign-in window of Collibra Console, for example 2024.01.0

5. Press `Enter` to confirm your selection.
6. If you have selected Repository, Jobserver and/or Collibra Console, enter the location where PostgreSQL 14.9 is installed.

   > Tip The default PostgreSQL 14.9 path on RHEL/Rocky/CentOS is **/usr/pgsql-14**.

7. Enter the necessary configuration for each of the selected services.
   » After the last configuration, the installation of the services automatically starts.

## Windows installation

1. Start the installer:
   Windows Server: **setup.bat**

   > Important The path of the installer file cannot contain spaces.

2. Enter the **Installation directory** and press `Enter`.
   The default location on Windows is **C:\collibra**
3. Enter the **Data directory** and press `Enter`.
   The default location on Windows is **C:\collibra_data**
4. Press `Enter` to each of the presented components that you want to install.
   If you don't want to install a specific component, press `n` followed by `Enter`.

   > Note We recommend that you have 4 dedicated nodes: one for the Data Governance Center service and Search service, one for Collibra Console, one for the Repository service, and one for Jobserver if applicable. Make sure that you have a fast network between the nodes.
   > Also make sure that you use the same installer version on all nodes. You can find the installer version of your environment at the bottom of the sign-in window of Collibra Console, for example 2024.01.0

5. Press `Enter` to confirm your selection.
6. Enter the necessary configuration for each of the selected services.
   » After the last configuration, the installation of the services automatically starts.

Below you find the configuration parameters for each of the services.

## DGC service configuration parameters

| Setting | Description |
|---|---|
| DGC port | The TCP port to access your Collibra DGC environment via your web browser.<br>The default port is *4400*. |
| DGC Shutdown port | The TCP port to stop the DGC service.<br>The default port is *4430*. |
| DGC minimum memory | The minimum amount of memory in megabytes for the DGC service.<br>This must be at least 1024 MB and no greater than 32 768 MB (32 GB). |
| DGC maximum memory | The maximum amount of memory in megabytes that can be assigned to the DGC service.<br>This must be at least 2048 MB and no greater than 32 768 MB (32 GB). |

## Repository service configuration parameters

| Setting | Description |
|---|---|
| Repository port | The TCP port to access the repository service. It is only used by the DGC service and the Collibra agent.<br>The default port is *4403*.<br>If you run multiple environments on one node, all ports must be unique for each environment. |
| Repository admin password (*) | The password that is used by the agent to access the Repository service. |
| Confirm repository admin password | The password as entered in the Repository admin password field. |
| Repository dgc password (*) | The password that is used by the DGC service to access the repository. |
| Confirm repository dgc password | The password as entered in the Repository dgc password field. |

| Setting | Description |
| --- | --- |
| Repository memory | The amount of memory for the Repository service in megabytes.<br>This must be at least 512 MB and no greater than 16 384 MB (16 GB). |

Note (*) These passwords can contain the following characters:

- lowercase letters
- uppercase letters
- numbers
- the following special characters: #?!@$%&*-

## Jobserver configuration parameters

| Setting | Description |
| --- | --- |
| Jobserver port | The TCP port to access the Jobserver service.<br>The default port is *4404*. |
| Jobserver database port | The TCP port to access the Jobserver database.<br>The default port is *4414*. |
| Jobserver monitoring port | The TCP port that is used by the monitoring service to monitor the Jobserver service.<br>The default port is *4424*. |
| Jobserver Spark monitoring port | The TCP port that is used by the monitoring service to monitor the Jobserver Spark service.<br>The default port is *4434*. |

## Search service configuration parameters

| Setting | Description |
|---|---|
| Search http port | The TCP port to access the Search service via REST API.<br>The default port is *4421*. |
| Search transport port | The TCP port for the communication between the DGC and the Search service.<br>The default port is *4422*. |
| Search memory | The amount of memory in megabytes that is assigned to the Search service.<br>The default value is *1024*. |

## Agent configuration parameters

| Setting | Description |
|---|---|
| Agent port | The TCP port that is used by Collibra Console to manage the services of an environment.<br>The default port is *4401*.<br>If you run multiple agents on one node, this port must be unique for each agent. |
| Node address | The hostname of the node on which the Agent service is running.<br>You cannot use a loopback address if you want to use the node in a multinode environment. |

## Collibra Console configuration parameters

| Setting | Description |
|---|---|
| Console port | The TCP port to access your Collibra Console via your web browser.<br>The default port is *4402*. |

| Setting | Description |
| --- | --- |
| Console database port | The TCP port to access the Collibra Console database. This is the database where the data and configuration of Collibra Console is stored.<br>The default port is *4420*. |

## What's next?

Create an environment.

# Create a CCSH environment

A Collibra Cloud Self-Hosted environment contains all the services that makes the Collibra platform operational for end-users. You can create multiple environments for different purposes, for example a development and production environment, but keep in mind that a service is always dedicated to one environment.

> Note With Collibra Console, you can manage many nodes, these are the services on which you have installed the Collibra services. These nodes must be installed with the same installer version as your Collibra Console.

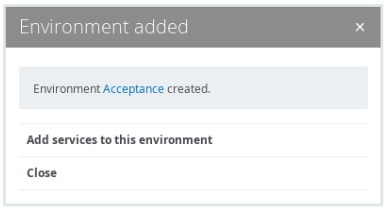
## Steps

1. Open Collibra Console with a user profile that has the **SUPER** role.
   » Collibra Console opens with the **Infrastructure** page.

   > Tip
   > - The default address to access Collibra Console is *<server hostname>:4402*, but you may have set another port during the installation of Collibra Console. Keep in mind that a firewall of your operating system can block the access to Collibra Console.
   > - The default credentials to sign in to Collibra Console are *Admin* / *admin*. We highly recommend that you edit the Collibra Console administrator's password after signing in for the first time.

2. In the tab pane, click **Add / Create**.
   » The **Add / Create** dialog box appears.

3. Click **Create environment**.
   » The **Create Environment** dialog box appears.
4. Enter a name.
5. Click **Create Environment**.




6. Perform one of the following steps:
   - Click **Close** to end the wizard.
   - Click **Add services to this environment** to immediately add services.

> Note If the node that hosts the service you want to add is not yet available in Collibra Console, click **Add services from a new node** under the drop-down list and add the node details.

# Add a node to a Collibra Cloud Self-Hosted environment

A node is a physical server that runs one or more services of a Collibra Data Intelligence Cloud environment.

## Prerequisites

- The node that you want to add to your infrastructure must be up and running and reachable from the Console that you are using.
- The version of the node must match the version of Collibra Console.

> Tip To add a node that was previously registered to another Collibra Console, go to the knowledge base on the Collibra Support Portal.

# Steps

1. Open Collibra Console with a user profile that has the **SUPER** role.
   » Collibra Console opens with the **Infrastructure** page.

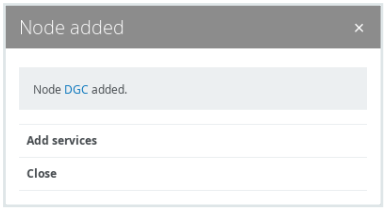
   Tip
   - The default address to access Collibra Console is *<server hostname>:4402*, but you may have set another port during the installation of Collibra Console. Keep in mind that a firewall of your operating system can block the access to Collibra Console.
   - The default credentials to sign in to Collibra Console are *Admin* / *admin*. We highly recommend that you edit the Collibra Console administrator's password after signing in for the first time.

2. In the tab pane, click **Add / Create**.
   » The **Add / Create** dialog box appears.
3. Click **Add node**.
   » The **Add node** dialog box appears.
4. Enter the necessary information.

| Field | Description |
|---|---|
| Node name | Enter a meaningful name for the node. |
| Hostname | Enter the hostname or IP address of the node, for example *192.168.1.100* or *repository-node-A*.<br>If you use a hostname, make sure that the Collibra Console can resolve the hostname.<br>Note Do not reuse hostnames, every hostname must be unique. If you reuse a hostname for a node that will be used in a repository cluster, the cluster won't synchronize. |
| Port | Enter the agent port. This is the port that you defined during the installation of the Agent on that node. The default value is 4401.<br>The agent port is the port through which Collibra Console connects to the node. |

5. Click **Add node**

6. Click **Close**.




## What's next?

Add services to your environment.

# Add a service to a Collibra Cloud Self-Hosted environment

A Collibra Data Intelligence Cloud on Collibra Cloud Self-Hosted environment consists of a collection of services, such as the DGC service and the Repository service. A service is hosted on a node. To add a service to an environment, the node must be added to the infrastructure that is managed by Collibra Console.

An operational Collibra environment requires at least the following services:

- Data Governance Center
- Repository
- Search
- Management Console.

> Note The Jobserver service is only required if you are ingesting data with Data Catalog. See Add a Jobserver to the Data Governance Center service for more information.

You can add a service to an environment in the following ways:

- Add a service via the global **Add / Create** button.
- Add a service via the environment details.
- Add services while creating an environment.

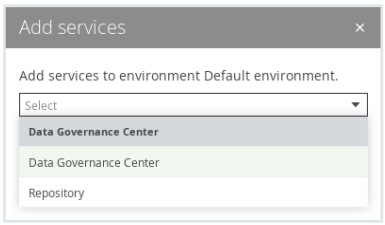
> Tip Make sure that the environment is stopped before adding services.

# Via global Add / Create button

1. Open Collibra Console with a user profile that has the **SUPER** role.
   » Collibra Console opens with the **Infrastructure** page.

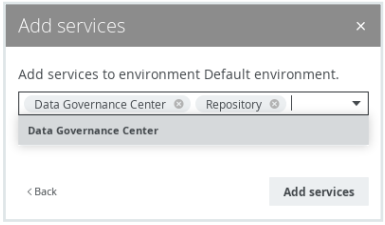
   Tip
   - The default address to access Collibra Console is *<server hostname>:4402*, but you may have set another port during the installation of Collibra Console. Keep in mind that a firewall of your operating system can block the access to Collibra Console.
   - The default credentials to sign in to Collibra Console are *Admin* / *admin*. We highly recommend that you edit the Collibra Console administrator's password after signing in for the first time.

2. In the tab pane, click **Add / Create**.
   » The **Add / Create** dialog box appears.
3. Click **Add services to environment / cluster**.
   » The **Select environment** dialog box appears.
4. Select the **Environment** option and select an environment from the drop-down list.
5. Click **Next**.
   » The **Add services** dialog box appears.
6. Click the relevant services in the drop-down list.

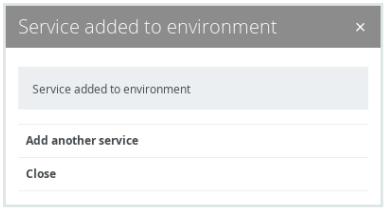



   Note If the node that hosts the service you want to add is not yet available in Collibra Console, click **Add services from a new node** under the drop-down list and add the node details.

7. Click **Add services**.

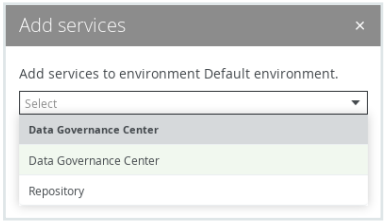


8. When you have added all services, click **Close**.



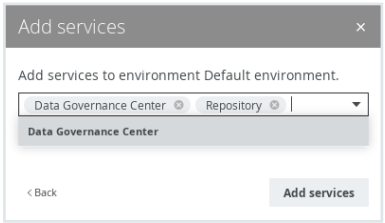

## Via environment details

1. Open Collibra Console with a user profile that has the **SUPER** role.
   » Collibra Console opens with the **Infrastructure** page.
2. In the tab pane, click the name of your environment.
   » The environment details appear.
3. Click **Add services**.
   » The **Add services** dialog box appears.
4. Click the relevant services in the drop-down list.

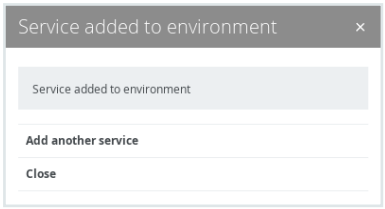



> Note If the node that hosts the service you want to add is not yet available in Collibra Console, click **Add services from a new node** under the drop-down list and add the node details.

5. Click **Add services**.



6. When you have added all services, click **Close**.




## What's next?

Start the environment.

# Start the environment

Starting an environment will start all services of the environment. You can also start the services individually, for more information, go to Start a service.

> Note When you use a network service account to start the agent and console services, the account must be available when the node starts. If the account is not available, the startup will fail.

## Steps

To start an environment, follow these steps:

1. Open Collibra Console with a user profile that has at least the **ADMIN** role.
   » Collibra Console opens with the **Infrastructure** page.
2. Click the name of an environment to show its details.
3. Click ▶ **Start**.

The environment is operational when all services are started and the environment has the status **running**.

# Configure SSL to access Collibra Cloud Self-Hosted

If you want to connect to Collibra Cloud Self-Hosted in a secure way with your web browser, you have to use SSL. This procedure explains how you can activate SSL access to CCSH.

> Tip For secure communication from CCSH to other services, for example an LDAP server, see Configure the SSL settings.

## Prerequisites

- You have knowledge of the JSON syntax.
- You have created a Java KeyStore according the procedure described by Oracle, for example **clientkeystore**.
- You have noted the following data while creating the Java KeyStore:
  - KeyStore file name: *clientkeystore* in the Oracle example.
  - KeyStore alias: *client* in the Oracle example.
  - KeyStore password: The password that you entered after executing the command of the first step in the Oracle example.
  - KeyStore alias password: The password that you entered as last step of step 2 in the Oracle example.
- You have stored the Java KeyStore on the server that hosts the DGC service in the **<collibra_data>/dgc/security** folder, for example **/opt/collibra_data/dgc/security**.

## Steps

To configure access to Collibra Cloud Self-Hosted over SSL, follow these steps:

1. Open a terminal session on the node on which the DGC service is installed.
2. Open the file **<collibra_data>/dgc/config/server.json** for editing.
3. Fill in the following parameters in the **httpsConnector** section:
   Add string values between double quotes.

| Parameter | Description |
|---|---|
| port | The port on which the HTTPS connector must bind. The value must be higher than 1024 to avoid root permissions.<br><br>Note If you want to use the default SSL port 443, you have to use a reverse proxy. |
| keyAlias | The KeyStore alias. |
| keyPass | The KeyStore alias password. |
| keystorePass | The KeyStore password. |
| keystoreFile | The full path to the KeyStore file name, for example **/opt/collibra_data/dgc/security/clientkeystore**. |

Example:

```
"httpsConnector" : {
    "port": 5404,
    "keyAlias": "your-alias",
    "keyPass": "your-password",
    "keystorePass": "your-password",
    "keystoreFile": "/opt/collibra_
data/dgc/security/collibradgc.jks" },
}
```

4. Save and close the file.
5. Open Collibra Console with a user profile that has the **SUPER** role.
   » Collibra Console opens with the **Infrastructure** page.
6. Open the DGC service settings for editing:
   a. Click the **General settings** section.
   b. Update the **Base URL** parameter with *https* and the new port.
7. Restart the environment.

Connect to your CCSH environment via the Base URL.

# Extra

To prevent regular HTTP traffic to Collibra Cloud Self-Hosted, update the **address** parameter with the value *127.0.0.1* in **<collibra_data>/dgc/config/server.json** and restart the environment.

This will not prevent the administration tools, for example Collibra Console, from connecting to CCSH without SSL.

For more information, go to the knowledge base on the Collibra Support Portal.

Chapter 4

# Upgrade to Collibra Cloud Self-Hosted

This section describes how to upgrade Collibra Data Governance Center to Collibra Cloud Self-Hosted.

# Upgrade requirements

Before you start the upgrade, you need all of the following information to ensure an easy, successful installation process. This section only focuses on the requirements of the core platform and does not take into account the connections to the data sources to ingest data.

## System requirements

### Supported Collibra Data Governance Center version

To upgrade to Collibra Cloud Self-Hosted, you need an on-premises Collibra environment 5.9.1. All older on-premises versions, that is 5.9.0 and older, must be first upgraded to 5.9.1 before you can upgrade to CCSH.

### Supported operating systems

If you want to upgrade to Collibra Data Intelligence Cloud on Collibra Cloud Self-Hosted, your current environment must be installed on one of the following operating systems:

- Red Hat Enterprise Linux/CentOS 7
- Red Hat Enterprise Linux/Rocky Linux 8
- Red Hat Enterprise Linux/Rocky Linux 9
- Windows Server 2016 2019, and 2022

> Important
> - Only the x86_64 architecture is supported.
> - On Linux, you can use root and standard users for the upgrade.
> - On Windows, you need administrative rights for the upgrade.

If your operating system of your current environment is not listed, then install a new Collibra environment on CCSH and restore the 5.9.1 backup.

## Hardware requirements

| Number of concurrent users | Number of assets | Recommended requirements for DGC service | Recommended requirements for Repository |
|---|---|---|---|
| 1 - 50 | < 1 million | 4 CPUs / 16 GB memory | 2 CPUs / 16 GB memory |
| 50 - 100 | 1 - 5 million | 4 CPUs / 16 GB memory | 4 CPUs / 16 GB memory |
| 50 - 200 | 5 - 25 million | 8 CPUs / 32 GB memory | 8 CPUs / 32 GB memory |
| 200 - 500 | 25 - 50 million | 16 CPUs / 64 GB memory | 16 CPUs / 64 GB memory |
| > 500 | > 50 million | 32 CPUs / 128 GB memory | 32 CPUs / 128 GB memory |

# Network requirements

Collibra Cloud Self-Hosted uses the following ports for the following services.

| Port | Default value | Purpose |
|---|---|---|
| Agent application | 4401 | TCP port used by Collibra Console to manage the services in a Collibra environment. |
| Console application | 4402 | TCP port to access your Collibra Console via your web browser. |
| Console database | 4420 | TCP port to access the database of Collibra Console. |
| Collibra Data Quality & Observability | 80 | TCP port to ingest Collibra DQ metadata over REST API. |

| Port | Default value | Purpose |
|---|---|---|
| DGC service, including Assessments, Usage Analytics, Privacy, Protect | 4400 | TCP port to access your Collibra environment via your web browser. |
| DGC shutdown port | 4430 | TCP port through which you can stop the DGC service. |
| Insights Data Access | 443 | TCP port to access Insights Data Access. |
| Jobserver database | 4414 | TCP port to access the Jobserver database. |
| Jobserver monitoring port | 4424 | Port used by the Monitoring service to monitor the Jobserver service. |
| Jobserver service | 4404 | TCP port to access the Jobserver service. |
| Jobserver Spark monitoring port | 4434 | Port used by the Monitoring service to monitor the Spark service. |
| Repository service | 4403 | TCP port to access the repository service. It is used only by the DGC service and the Collibra agent. |
| Search HTTP port | 4421 | TCP port to access the Search service. |
| Search Transport port | 4422 | TCP port used by the DGC service to communicate with the Search service. |

## Other

- You have enough free disk space in the volume that hosts the data folder, **collibra_data**. The free disk space must be at least the size of the current data.
  For example, if your data in the data folder takes 5 GB, you need at least 5 GB of free disk space on that volume to upgrade.

- The status of your repository is Green or Orange.

  > Warning If the status of the repository is Red, do not start the upgrade procedure. Contact Collibra Support.

- You must use the same user account that you used to install Collibra Data Governance Center, to perform the upgrade to Collibra Cloud Self-Hosted. If the user account is no longer active, go to Upgrade an environment with another user account in the Troubleshooting section or install a new environment and restore a backup.

# Prepare your environment for upgrade

## General preparation

You have downloaded the Collibra Cloud Self-Hosted installer on all nodes of your environment.

## Does your environment have repo clusters?

If your environment has repo clusters, then do the following before starting the upgrade:

1. Delete all repo services from the repository cluster.
2. Start all repo services that were used as replica.
3. Stop all repo services again.

> Tip This is not needed for the master repo service.

# Upgrade to Collibra Cloud Self-Hosted

This section describes how you can upgrade an on-premises Collibra Data Governance Center 5.9.1 to Collibra Cloud Self-Hosted 2024.01.

> Important If your environment is configured with repository clusters, make sure that you have executed this procedure before you start the upgrade.

# Steps

## Upgrade to CCSH on Linux

1. Stop the environment:
   a. Open Collibra Console with a user profile that has at least the **ADMIN** role.
      » Collibra Console opens with the **Infrastructure** page.
   b. Click the environment that you want to stop.
   c. Click ■ **Stop**.
      » The **Stop environment** dialog box appears.
   d. Click **Stop environment**.
   e. Wait until all the nodes of the environment have the status **Stopped**.
   f. Stop Collibra Console and the Collibra DGC Agent. In the terminal of the node that runs Collibra Console:
      a. As root:

```
service collibra-agent stop
service collibra-console stop
```

      b. Manual stop of the services:

```
/opt/collibra/console/bin/console stop
/opt/collibra/agent/bin/agent stop
```

2. Start the installer and follow the instructions.

> Note
> - The path to your PostgreSQL 14.9 installation differs per Linux operating system. In the following example, the path is of a default installation on Red Hat, Rocky Linux, or CentOS.
> - The amount of time it takes to upgrade your environment depends on the size of your repository. The larger the database, the more time it takes to upgrade.

```
# ./dgc-linux-2024.01.0.sh
Verifying archive integrity...  100%   All good.
Uncompressing DGC Installer  100%
Specify the installation directory [/opt/collibra]:

/opt/collibra contains a previous installation. Do you want
to perform an update? [y/N]
y
Before you can schedule an upgrade, you need to:
 - Create a backup of the entire environment.
 - In Collibra Console, stop all running services on this
node.
Have you completed these steps? [yes,NO]

yes
2023-10-16 10:32:51.581 - SUCCESS - Check umask settings
2023-10-16 10:32:51.592 - SUCCESS - Create installation and
data directories
2023-10-16 10:32:51.597 - SUCCESS - Check Search system
requirements
2023-10-16 10:32:52.030 - SUCCESS - Remove Agent system
service
2023-10-16 10:32:52.280 - SUCCESS - Remove Console system
service
2023-10-16 10:32:54.773 - SUCCESS - Extract JRE
2023-10-16 10:32:54.893 - SUCCESS - Copy server files
2023-10-16 10:32:54.894 - SUCCESS - Create PostgreSQL
temporary directories
2023-10-16 10:32:54.896 - SUCCESS - Move old postgresql
files to temporary directory
2023-10-16 10:32:54.897 - SUCCESS - Install PostgreSQL
configuration files to temporary directory
2023-10-16 10:33:12.658 - SUCCESS - Update postgresql
2023-10-16 10:33:12.706 - SUCCESS - Update postgresql path
in configuration files
2023-10-16 10:33:12.711 - SUCCESS - Remove tracing
directories
2023-10-16 10:33:12.790 - SUCCESS - Prepare for jobserver
update
2023-10-16 10:33:12.790 - SUCCESS - Create spark
directories
2023-10-16 10:33:14.324 - SUCCESS - Extract Spark
2023-10-16 10:33:14.333 - SUCCESS - Replace variables in
spark files
2023-10-16 10:33:14.342 - SUCCESS - Set permissions on
Spark directory
```

```
...

2023-10-16 10:33:36.422 - SUCCESS - Create installation
configuration file
2023-10-16 10:33:36.468 - SUCCESS - Create uninstall script
2023-10-16 10:33:36.473 - SUCCESS - Cleanup temporary
directories
2023-10-16 10:33:36.474 - SUCCESS - Cleanup old Postgres
binaries
2023-10-16 10:33:36.679 - SUCCESS - Make sure all
permissions are correct
2023-10-16 10:33:37.089 - SUCCESS - Install Agent system
service
2023-10-16 10:33:37.312 - SUCCESS - Install Console system
service
2023-10-16 10:33:41.431 - SUCCESS - Start Agent
2023-10-16 10:33:45.620 - SUCCESS - Start Console
2023-10-16 10:33:45.620 - COMPLETED - Installation finished
in 54352ms.
```

Note
- If you have a multi-node installation, repeat this step on every node of the environment until you have upgraded all nodes.
- If you have a node that only runs the Monitoring service, you don't have to upgrade that node.

3. Start the environment:
   a. Open Collibra Console with a user profile that has at least the **ADMIN** role.
      » Collibra Console opens with the **Infrastructure** page.
   b. Click the name of an environment to show its details.
   c. Remove the Monitoring service from the environment if it is still available. If this service was installed on a dedicated node, you can remove that node too.

      Tip If the Monitoring service was installed together with other services, the Monitoring service is automatically removed during the upgrade.

   d. Click ▶ **Start**.
4. Reindex Collibra.

# Upgrade to CCSH on Windows

1. Stop the environment.
   a. Open Collibra Console with a user profile that has at least the **ADMIN** role.
      » Collibra Console opens with the **Infrastructure** page.
   b. Click the environment that you want to stop.
   c. Click ■ **Stop**.
      » The **Stop environment** dialog box appears.
   d. Click **Stop environment**.
   e. Wait until all the nodes of the environment have the status **Stopped**.
   f. Stop Collibra Console and the Agent via the Windows services window. The services to be stopped are **Agent** and **Management Console**.
2. In the command prompt, start the installer: setup.bat.

> Note
> - The amount of time it takes to upgrade your environment depends on the size of your repository. The larger the database, the more time it takes to upgrade.

```
# ./setup.bat
Verifying archive integrity...  100%   All good.
Uncompressing DGC Installer  100%
Specify the installation directory [C:\collibra]:

C:\collibra contains a previous installation. Do you want
to perform an update? [y/N]
y
Before you can schedule an upgrade, you need to:
- Create a backup of the entire environment.
- In Collibra Console, stop all running services on this
node.
Have you completed these steps? [yes,NO]

yes
2023-10-16 10:32:51.592 - SUCCESS - Create installation and
data directories
2023-10-11 11:43:17.841 - SUCCESS - Stop Agent
2023-10-16 10:32:52.030 - SUCCESS - Remove Agent system
service
2023-10-11 11:43:22.368 - SUCCESS - Stop Console
2023-10-16 10:32:52.280 - SUCCESS - Remove Console system
```

```
service
2023-10-16 10:32:54.773 - SUCCESS - Extract JRE
2023-10-16 10:32:54.893 - SUCCESS - Copy server files

...

2023-10-16 10:33:36.422 - SUCCESS - Create installation
configuration file
2023-10-16 10:33:36.468 - SUCCESS - Create uninstall script
2023-10-16 10:33:36.473 - SUCCESS - Cleanup temporary
directories
2023-10-11 11:45:24.755 - SUCCESS - Cleanup obsolete
installation files
2023-10-11 11:45:24.757 - SUCCESS - Cleanup legacy
installation files
2023-10-16 10:33:36.474 - SUCCESS - Cleanup old Postgres
binaries
2023-10-16 10:33:37.089 - SUCCESS - Install Agent system
service
2023-10-16 10:33:37.312 - SUCCESS - Install Console system
service
2023-10-16 10:33:41.431 - SUCCESS - Start Agent
2023-10-16 10:33:45.620 - SUCCESS - Start Console
2023-10-16 10:33:45.620 - COMPLETED - Installation finished
in 54352ms.
```

Note
- If you have a multi-node installation, repeat this step on every node of the environment until you have upgraded all nodes.
- If you have a node that only runs the Monitoring service, you don't have to upgrade that node.

3. Start the environment:
   a. Open Collibra Console with a user profile that has at least the **ADMIN** role.
      » Collibra Console opens with the **Infrastructure** page.
   b. Click the name of an environment to show its details.
   c. Remove the Monitoring service from the environment if it's still available. If this service was installed on a dedicated node, you can remove that node too.

      Tip If the Monitoring service was installed together with other services, the Monitoring service is automatically removed during the upgrade.

d. Click ▶ **Start**.

4. Reindex Collibra.

# Unattended installation and upgrade

Instead of following the installation wizard, you can also install or upgrade the software in an unattended way by executing the installation command in combination with a configuration file.

This allows you to automate the installation process on various servers.

# Unattended installation of Collibra Cloud Self-Hosted

In this section, you learn how you can install Collibra Cloud Self-Hosted without manual interaction.

## Prerequisites

- The prerequisites of a normal installation.
- A valid configuration file in JSON format.
  None of the configuration parameters is required. For every parameter that is not provided, the system will use a default value.

# Steps

## Unattended installation of CCSH on Linux

1. Open a terminal session and go to the directory with the installer.
2. Run the following command:
   - As root: `sudo ./dgc-linux-2024.01.0.sh -- --config /full-path/to/config`
   - As non-root: `./dgc-linux-2024.01.0.sh -- --config /full-path/to/config`

> Tip
> - You can replace `--config` by `-c`.
> - Use the full path to the configuration file, even if it is in the same directory as the installer.

Example output:

```
~$ ./dgc-linux-2024.01.0.sh -- --config
/home/johndoe/Downloads/config.json
Verifying archive integrity... 100% All good.
Uncompressing DGC Installer 100%
10:49:02.235 - Using configuration file :
/home/johndoe/Downloads/config.json
10:49:02.324 - SUCCESS - Check umask settings
10:49:02.326 - SUCCESS - Create installation and data directories
10:49:02.353 - SUCCESS - Create installation configuration file
10:49:02.454 - SUCCESS - Create uninstall script.
10:49:04.559 - SUCCESS - Extract JRE

...

10:49:20.826 - SUCCESS - Start Agent
10:49:24.464 - SUCCESS - Start Console
10:49:24.464 - Installation finished in 22184ms.
```

## Unattended installation of CCSH on Windows

1. Open a command-line session (Command Prompt or Windows PowerShell) as Administrator and go to the directory with the installer.
2. Run the following command: `setup.bat --config <full-path/to/config>`

> Tip
> - You can replace `--config` by `-c`.
> - Use the full path to the configuration file, even if it is in the same directory as the installer.

Example output:

```
~$ ./setup.bat -- --config C:\Users\johndoe\Downloads\config.json
Verifying archive integrity... 100% All good.
Uncompressing DGC Installer 100%
10:49:02.235 - Using configuration file :
C:\Users\johndoe\Downloads\config.json
10:49:02.324 - SUCCESS - Check umask settings
10:49:02.326 - SUCCESS - Create installation and data directories
10:49:02.353 - SUCCESS - Create installation configuration file
10:49:02.454 - SUCCESS - Create uninstall script.
10:49:04.559 - SUCCESS - Extract JRE

...

10:49:20.826 - SUCCESS - Start Agent
10:49:24.464 - SUCCESS - Start Console
10:49:24.464 - Installation finished in 22184ms.
```

# What's next?

After you have installed all the services, create an environment.

# Unattended upgrade to Collibra Cloud Self-Hosted

Similar to an unattended installation, you can also upgrade the software in an unattended way.

## Prerequisites

- The prerequisites of a normal upgrade.

> Note If you use a configuration file, you can edit the necessary parameters of your existing services, for example to edit a TCP port.

## Steps

### Unattended upgrade to CCSH on Linux

1. Open a terminal session.
2. Go to the directory with the installer.
3. Run the following command:

> Tip
> - If you use a configuration file, for example to add or reconfigure a service, you can replace `--upgrade-config` by `-uc`.
> - Use the full path to the configuration file, even if it is in the same directory as the installer.

| OS | Command |
|---|---|
| Linux (root) | ◦ Upgrade with the default options:<br>`sudo ./dgc-linux-2024.01.0.sh -- \`<br>`--upgrade /path/to/installation`<br>◦ Upgrade with a configuration file:<br>`sudo ./dgc-linux-2024.01.0.sh -- \`<br>`--upgrade /path/to/installation \`<br>`--upgrade-config /path/to/config` |
| Linux (standard user) | ◦ Upgrade with the default options:<br>`./dgc-linux-2024.01.0.sh -- \`<br>`--upgrade /path/to/installation`<br>◦ Upgrade with a configuration file:<br>`./dgc-linux-2024.01.0.sh -- \`<br>`--upgrade /path/to/installation \`<br>`--upgrade-config /path/to/config` |

## Unattended upgrade to CCSH on Windows

1. Open the command prompt.
2. Go to the directory with the installer.
3. Run one the following commands:

> Tip
> - If you use a configuration file, for example to add or reconfigure a service, you can replace `--upgrade-config` by `-uc`.
> - Use the full path to the configuration file, even if it is in the same directory as the installer.

   - Upgrade with the default options:
     ```
     setup.bat --upgrade <drive>:\path\to\installation
     ```
   - Upgrade with a configuration file:
     ```
     setup.bat --upgrade <drive>:\path\to\installation \
     --upgrade-config <drive>:\path\to\config
     ```

## What's next?

All Collibra services will be upgraded and be readily available upon the upgrade completion.

# Add extra service during an unattended upgrade

When you upgrade a node in an unattended way, you can add extra services on that node or change the configuration of an existing service. To do this, you have to create a new configuration file that includes at least the key `componentSet` and as value the list of services that you want to add or update, for example:

```
{"componentSet":["SEARCH","CONSOLE"]}
```

If you don't want to use the default parameters for the added service(s), you also have to add the configuration key-value pairs for each service.

Adding a service to a node or reconfiguring a service requires you to add an extra parameter (`--upgrade-config` or `-uc`) to the upgrade command.

> Note For an upgrade to CCSH 2024.01, you need Collibra Data Governance Center 5.9.1.

Example Search service configuration:

```
{
    "componentSet" : ["SEARCH","CONSOLE"],
    "searchHttpPort" : 4421,
    "searchTransportPort" : 4422,
    "searchMemory" : 1024,
    "postgresqlPath" : "/usr/pgsql-14",
    "consolePort" : 4402,
    "consoleDatabasePort" : 4420
}
```

# Unattended installation configuration parameters

The following table contains the parameters that you can use in the JSON installation file for an unattended installation of Collibra Cloud Self-Hosted. If the parameter is not provided, a default value is used.

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| installationDirectory | Name of the directory where Collibra DGC will be installed.<br>On Windows, the directory must have a URL format (file:///path). | string | • Default (Linux with root permission): */opt/collibra*<br>• Default (Linux without root permission): **/home/<user>/collibra** | Default: *file:///c:/collibra* |
| dataDirectory | Name of the directory where the Collibra data will be stored.<br>On Windows, the directory must have a URL format (file:///path). | string | • Default (Linux with root permission): */opt/collibra_data*<br>• Default (Linux without root permission): **/home/<user>/collibra_data** | Default: *file:///c:/collibra_data* |
| repositoryMemory | Reserved random access memory in MB for the repository service. | int | Default value: *1024* | Default value: *1024* |
| dgcMinMemory | Minimum amount of memory in MB for the DGC service. | int | Default value: *1024* | Default value: *1024* |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| dgcMaxMemory | Maximum amount of memory in MB for the DGC service. | int | Default value: *2048* | Default value: *2048* |
| dgcPort | TCP port to access the DGC service. | long int | Default value: *4400* | Default value: *4400* |
| dgcShutdownPort | TCP port to shut down a Collibra DGC environment. | long int | Default value: *4430* | Default value: *4430* |
| repositoryPort | TCP port to access the repository database. | long int | Default value: *4403* | Default value: *4403* |
| consolePort | TCP port to access Collibra Console. | long int | Default value: *4402* | Default value: *4402* |
| consoleDatabasePort | TCP port to access the Collibra Console database. | long int | Default value: *4420* | Default value: *4420* |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| consoleDatabasePassword | Password used by Collibra Console to store data in its database. | string | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. |
| consoleDatabaseAdminPassword | Password to directly access the Collibra Console database. | string | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. |
| agentPort | TCP port that is used by Collibra Console to connect to the Collibra agent for management purposes. | long int | Default value: *4401* | Default value: *4401* |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| jobserverPort | TCP port to access the Jobserver. | long int | Default value: *4404* | Default value: *4404* |
| jobserverDatabasePort | TCP port to access the Jobserver database. | long int | Default value: *4414* | Default value: *4414* |
| searchHttpPort | TCP port to access the Search service via REST API | long int | Default value: *4421* | Default value: *4421* |
| searchTransportPort | TCP port for the communication between the DGC and Search service. | long int | Default value: *4422* | Default value: *4422* |
| searchMemory | The memory in MB assigned to the Search service. | int | Default value: *1024* | Default value: *1024* |
| nodeHostName | The hostname of the node on which you are installing services.<br>If you are installing a multinode environment, you have to use this parameter with another name than *localhost*. | string | Default value: *localhost*<br>If you use this default value, the node cannot be used in multinode environments. | Default value: *localhost*<br>If you use this default value, the node cannot be used in multinode environments. |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| repoAdminPassword | Admin password to access the repository database directly.<br>This should only be done by experienced database administrators. | string | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. |
| repoDgcPassword | Password for the DGC service to obtain access to the repository database. | string | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. | There is no default value but you have to fill in a password if you add this parameter. Empty strings are not allowed.<br>If you don't add this parameter, the password will be automatically generated. |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| componentSet | List of services to install:<br>• DGC<br>• REPOSITORY<br>• JOBSERVER<br>• AGENT<br>• CONSOLE<br>• SEARCH<br>Tip If you install DGC, REPOSITORY, SEARCH and/or JOBSERVER, the AGENT is automatically included. | string | Example: *DGC,CONSOLE* | Example: *DGC,CONSOLE* |
| initDaemon | Select a custom init daemon:<br>• systemv<br>• upstart<br>• systemd<br>This is a Linux only parameter.<br>Be careful when you specify an init daemon, it may result in an unstable operating system. | int | The default value is the one that is the most appropriate for your Linux system. | Not applicable |

| Parameter | Description | Type | Linux example | Windows example |
|---|---|---|---|---|
| userName | The name of the user who will install the software.<br>This is only required if the userGroup is not the same as the userName.<br>This is a Linux only parameter. | string | The default value is the one that is used to execute the installation command. | Not applicable |
| userGroup | The group to which the user belongs.<br>This is only required if the userGroup is different from the userName.<br>This is a Linux only parameter. | string | The default value is the same as the userName. | Not applicable |
| postgresqlPath | The name of the directory where PostgreSQL 14.9 is installed. | string | Default: */usr/pgsql-14.9* | Not applicable |

Note
- Ensure that you add the escape character ( \ ) in the Windows paths in front of a backslash.
  Example: **C:\\collibra_data**
- Only use double quotes in the configuration file.

## Example input file

Next you find an example JSON file for an unattended installation.

```
{
   "installationDirectory" : "/home/johndoe/collibra/",
   "dataDirectory" : "/home/johndoe/collibra_data/",
   "postgresqlPath" : "/usr/pgsql-14.9",
   "repositoryMemory" : 1024,
   "dgcMinMemory" : 1024,
   "dgcMaxMemory" : 2048,
   "dgcPort" : 4400,
   "dgcShutdownPort" : 4430,
   "repositoryPort" : 4403,
   "consolePort" : 4402,
   "consoleDatabasePort" : 4420,
   "agentPort" : 4401,
   "jobserverPort" : 4404,
   "jobserverDatabasePort" : 4414,
   "searchHttpPort" : 4421,
   "searchTransportPort" : 4422,
   "searchMemory" : 1024,
   "repoAdminPassword" : "aV3r4Str0ngP@sww0rd",
   "repoDgcPassword" : "aV3r4Str0ngP@ssw0rd",
   "userName" : "johndoe",
   "userGroup" : "johndoe",
   "initDaemon" : null,
   "componentSet" : [ "CONSOLE", "JOBSERVER", "AGENT",
"REPOSITORY", "DGC", "SEARCH" ]
}
```

# Monitoring your Collibra Cloud Self-Hosted environment

If you are running Collibra Cloud Self-Hosted environments, we cannot monitor their health like we can with cloud environments. However, you can set up your own monitoring systems to ingest data, alert, and visualize the usage of your environment.

In this section, we describe what and how you can look up the monitoring data.

# System metrics

You can monitor the following system metrics in your environment:

- CPU usage
- Memory, used and available
- File system, used and available

> Tip We recommend setting an alert when file system usage is above 80%.

If you are using services or daemons, we recommend monitoring the health of the collibra-agent and collibra-console services.

The following is an example configuration for OpenTelemetry collector to collect basic system metrics. This may be different for the observability tool that you are using.

```
receivers:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector-con-
trib/tree/main/receiver/hostmetricsreceiver
  hostmetrics:
    scrapers:
      cpu:
      memory:
      disk:
      load:
      paging:
      processes:
      network:
      filesystem:

processors:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-col-
lector/tree/main/processor/batchprocessor
  batch:

  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector-con-
trib/tree/main/processor/resourcedetectionprocessor
  resourcedetection:
    detectors:
      - env
      - system
```

```
exporters:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector/tree/main/exporter/debugexporter
  debug:

  # Add appropriate exporter(s) here
  # See all available ones: https://github.com/open-tele-
metry/opentelemetry-collector-contrib/tree/main/exporter

service:
  pipelines:
    metrics:
      receivers:
        - hostmetrics
      processors:
        - resourcedetection
        - batch
      exporters:
        - debug
        # Add appropriate exporter(s) here
```

# Importing logs in an observability tool

You can open log files of all Collibra services in Collibra Console. Instead of analyzing log files in Collibra Console, you can also collect them in the observability tool of your choice. To know where to find the log files and what the content is, go to the Platform configuration section.

The path to the log files is typically **/path/to/collibra_data/<service name>/logs**.

The following is an example configuration for OpenTelemetry collector to collect the logging of the Data Governance Center service. This may be different for the observability tool that you are using.

```
receivers:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector-con-
trib/tree/main/receiver/filelogreceiver
  filelog/collibra_dgc:
    include:
      - ${env:COLLIBRA_DATA_PATH}/dgc/logs/dgc.log
```

```
    resource:
      service.name: collibra-dgc
    multiline:
      line_start_pattern: \d{4}-\d{2}-\d{2}\s\d{2}:\d{2}:\d
{2}.\d{3}

  # Add more filelog receivers for each log that you want to
collect

processors:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-col-
lector/tree/main/processor/batchprocessor
  batch:

  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector-con-
trib/tree/main/processor/resourcedetectionprocessor
  resourcedetection:
    detectors:
      - env
      - system

exporters:
  # See more details: https://github.com/open-tele-
metry/opentelemetry-collector/tree/main/exporter/debugexporter
  debug:

  # Add appropriate exporter(s) here
  # See all available ones: https://github.com/open-tele-
metry/opentelemetry-collector-contrib/tree/main/exporter

service:
  pipelines:
    logs:
      receivers:
        - filelog/collibra_dgc
        # Add more filelog receivers for each log that you want
to collect
      processors:
        - resourcedetection
        - batch
      exporters:
        - logging
        # Add appropriate exporter(s) here
```

# Environment health check

You can see the health of your environment infrastructure in Collibra Console indicated by a colored disc in front of the infrastructure element.

We also provide a REST API to monitor all the infrastructure elements.

| Infra-structure element | API documentation | Example response |
|---|---|---|
| Envir-onment | https://<your-console-url>/-doc-s/rest/in-dex.html#/environment/findAll_1 | Example - Environment details https://<your-console-url>/rest/environment/ [{ "cre-atedDate":1699578732.344000-000,<br>"mod-ifiedDate":1699578732.34400-0000, "id":"<uuid of environment>", "name":"Default envir-onment", "serviceIdSet":["<uuid service 1>","<uuid service 2>","<uuid service 3>","<uuid service 4>"], "status":"RUNNING" }] |

| Infra-structure element | API documentation | Example response |
|---|---|---|
| Service | https://<your-console-url>/-docs/rest/index.html#managed-services/findAll_2 | Example - Services details<br>https://<your-console-url>/rest/service<br>[<br>{<br>"cre-atedDate":1699578732.134000-000,<br><br>"mod-ifiedDate":1699578732.13400-0000,<br>"id":"<uuid of service",<br>"nodeId":"<uuid of node>",<br>"status":"RUNNING",<br>"errorMessage":"",<br>"requiredDependencies":<br>["SEARCH","REPOSITORY"],<br>"optionalDependencies":<br>["SPARK"],<br>"type":"DGC"<br>},<br>{<br>"cre-atedDate":1699578732.118000-000,<br><br>"mod-ifiedDate":1699578732.11800-0000,<br>"id":"<uuid of dgc service>",<br>"nodeId":"<uuid of node>",<br>"status":"RUNNING",<br>"errorMessage":"",<br>"requiredDependencies":[],<br>"optionalDependencies":[],<br>"type":"SPARK"<br>},<br>{<br>"cre-atedDate":1699578732.076000-000, |

| Infra-structure element | API documentation | Example response |
|---|---|---|
| | | "mod-ifiedDate":1699578732.07600-0000,<br>"id":"<uuid of search service>",<br>"nodeId":"<uuid of node>",<br>"status":"RUNNING",<br>"errorMessage":"",<br>"requiredDependencies":[],<br>"optionalDependencies":[],<br>"type":"SEARCH"<br>},<br>{<br>"cre-atedDate":1699578732.098000-000,<br><br>"mod-ifiedDate":1699578732.09800-0000,<br>"id":"<uuid of repo service>",<br>"nodeId":"<uuid of node>",<br>"status":"RUNNING",<br>"errorMessage":"",<br>"requiredDependencies":[],<br>"optionalDependencies":[],<br>"type":"REPOSITORY"}<br>] |

| Infra-structure element | API documentation | Example response |
|---|---|---|
| Node | • https://<your-console-url>/-doc-s/rest/index.html#/node/findAll_3<br>• https://<your-console-url>/-doc-s/rest/index.html#/node/status | • Example 1 - Node details https://<your-console-url>/rest/node<br><br>`[{ "cre-atedDate":1699578732.031-000000, "mod-ifiedDate":1699578733.67-0000000, "id":"<uuid of node>", "hostName":"localhost", "port":4401, "name":"Default node", "managedServiceIdSet": ["<uuid of service 1 on this node>", "<uuid of service 2 on this node>", "<uuid of service 3 on this node>", "<uuid of service 4 on this node>"] }]`<br><br>• Example 2 - Status of a node https://<your-console-url>/rest/node/<node uuid>/status<br><br>`"UP"` |