Cloudera Data Engineering 1.5.4

Managing virtual clusters

Date published: 2020-07-30
Date modified: 2024-05-30

CLOUD∃RA

Legal Notice

Contents

Creating virtual clusters

In Cloudera Data Engineering (CDE), a virtual cluster is an individual auto-scaling cluster with defined CPU and memory ranges. Jobs are associated with virtual clusters, and virtual clusters are associated with an environment. You can create as many virtual clusters as you need.

Before you begin

To create a virtual cluster, you must have an environment with Cloudera Data Engineering (CDE) enabled.

Procedure

1. In the Cloudera Data Platform (CDP) console, click the Data Engineering tile. The CDE Home page displays.
2. Click Administration in the left navigation menu, select the environment you want to create a virtual cluster in.

3.
In the Virtual Clusters column, click ⊕ at the top right to create a new virtual cluster.

If the environment has no virtual clusters associated with it, the page displays a Create DE Cluster button that launches the same

Administration / Create a Virtual Cluster

Cluster Name *

    Cluster Name

Service *

    service-01 (cluster-czct6k5g)   ▼

What do you intend to use this Virtual Cluster for? *

    ◉ Core (Tier 1)   ○ All Purpose (

       Operational deployment & monitoring of batch   Develop using
       based transformations   both batch & st

Unsure which one to select? Review the feature comparison

Spark Version

    Spark 3.2.3   ▼

Resource Pooling and Capacity ⓘ

Set the Guaranteed and Maximum number of resources your users can use in this Virtual Cluster as a whole. Guaranteed ensures users can use these resources for spark executors being preem

Note: CDE Infrastructure requires a minimum of 8 CPU cores and 24 GB in memory

CPU (cores) *

Guaranteed ⓘ   Maximum ⓘ

    8   200

A Maximum capacity of 200 cores CPU can be set so that this cluster may scale under high demand situations

Memory (GB) *   **6**

Guaranteed ⓘ   Maximum ⓘ

4. Enter a Cluster Name.

   Cluster names must include the following:

   - Begin with a letter
   - Be between 3 and 30 characters (inclusive)
   - Contain only alphanumeric characters and hyphens

5. Select the CDE Service to create the virtual cluster in.

   The environment you selected before launching the wizard is selected by default, but you can use the wizard to create a virtual cluster in a different CDE service.

6. Select one of the following CDE cluster types:

   - Core (Tier 1): Batch-based transformation and engineering options include:

     - Autoscaling Cluster
     - SDX/Lakehouse
     - Job Lifecycle
     - Monitoring
     - Workflow Orchestration

   - All Purpose (Tier 2) - Develop using interactive sessions and deploy both batch and streaming workloads. This option includes all options in Tier 1 with the following:

     - Shell Sessions - CLI and Web
     - JDBC/SparkSQL (Coming soon)
     - IDE (Coming Soon)

7. In Capacity, specify the guaranteed and the maximum number of CPU cores, GPU cores, and Memory in gigabytes to configure elastic quota. The cluster can utilize resources up to the maximum set capacity to run the submitted Spark applications.

   You can get a minimum guaranteed and maximum capacity of resources (CPU and memory) using guaranteed quota and maximum quota. The guaranteed quota dictates the minimum amount of resources available for allocation for a VC at all times. The resources above the guaranteed quota and within the VC's maximum quota can be used by any VC on demand if the cluster capacity allows for it.

   GPU (Technical Preview): You can set the guaranteed and maximum GPU resource quota for this virtual cluster for Spark 3 jobs to use.

   Elastic quotas allow the VC to acquire unused capacity in the cluster when their guaranteed quota limit gets exhausted. This ensures efficient use of resources in the cluster. At the same time, the maximum quota limits the threshold amount of resources a VC can claim in the cluster at any given time.

   For information about configuring resource pool and capacity, see Managing cluster resources using Quota Management.

8. Select the Spark Version to use in the virtual cluster.

9. Under Retention (Preview), click Enable Job Run and Log Retention Policy to configure the job run and log retention policy. The retention policy lets you specify how long to retain the job runs and logs, after which it will be deleted to save storage costs and improve performance. By default, in CDE there is no expiration period and both job runs and logs are retained forever. Provide the following to configure the duration:

   a) Enter Value: Enter a whole number greater than zero to set the duration. Ensure there are no decimals or other characters.

    b) Select Period: Select Hours, Days, or Weeks from the drop-down list to set the period of time for which job runs and logs are to be retained.

> ⚠️ **Important:** When you edit the log retention policy configuration, you must restart the runtime-api-server pod using the `kubectl rollout restart deployment/<deployment-name> -n <namespace>` command to apply the changes. The namespace is the VC ID found on the virtual cluster details page.
>
> For example:
>
> ```
> kubectl rollout restart deployment/dex-app-fww6lrgm-api -n dex-a
> pp-fww6lrgm
> ```

10. Click Configure Email Alerting if you want to receive notification mails. Provide the following email configuration information:

- Sender email address.
- Your mail server hostname and port.
- The username and password of the email user who will be logged into the mail server as the sender of the alert emails.
- Select a secure connection method to be used when communicating with the SMTP server.
- Click Test SMTP Configs to test the configurations set for SMTP. This helps you to test the SMTP configuration before creating the cluster.

11. Click Create.

Results

On the CDE Home page, select the environment to view the virtual cluster initialization status. You can also click the three-dot menu for the virtual cluster to view the logs.

What to do next

You must initialize each virtual cluster you create and configure users before creating jobs.

Cloudera Data Engineering provides a suite of example jobs with a combination of Spark and Airflow jobs, which include scenarios such as reading and writing from object storage, running an Airflow DAG, and expanding on Python capabilities with custom virtual environments. For information about running example jobs, see *CDE example jobs and sample data*.

Related Information
Resource scheduling and management

Initializing virtual clusters

You must initialize the cluster by performing manual steps for each virtual cluster you create.

Procedure

1. Download cde-utils.sh to your local machine.

2. Create a directory to store the files, and change to that directory:

```
mkdir -p /tmp/cde-latest && cd /tmp/cde-latest
```

3. Copy the script

Embedded Container Service (ECS)

Copy the extracted utility script (cde-utils.sh) to the Embedded Container Service (ECS) cluster host which has the ECS Master installed. To identify the ECS cluster hosts:

   a. Log in to the Cloudera Manager web interface.
   b. Go to Clusters Experience Cluster ECS Hosts .
   c. Select the host which has the ECS Master installed, and copy the script to that host.

Red Hat OpenShift Container Platform (OCP)

Copy the extracted utility script (cde-utils.sh) and the OpenShift kubeconfig file to one of the HDFS service gateway hosts, and install the kubectl utility:

   • Log in to the Cloudera Manager web interface.
   • Go to Clusters Base Cluster HDFS Instances .
   • Select one of the Gateway hosts, log in using the security password that was set, and copy the script to that host.
   • Copy the OCP kubeconfig file to the same host.
   • Export the OCP kubeconfig file:

   ```
   export KUBECONFIG=[***path_of_the_copied_OCP_Kubeconfig_file***]
   ```

   • On that host, install the kubectl utility following the instructions in the Kubernetes documentation. Cloudera recommends installing the version that matches the Kubernetes version installed on the OpenShift cluster.

4. On the cluster host that you copied the script to, set the script permissions to be executable:

   ```
   chmod +x /path/to/cde-utils.sh
   ```

5. Identify the virtual cluster endpoint:

   a. In the Cloudera Manager web UI, go to the Experiences page, and then click Open CDP Private Cloud Experiences.
   b. Click the Data Engineering tile.
   c. Select the CDE service containing the virtual cluster you want to activate.
   d. Click Cluster Details.
   e. Click JOBS API URL to copy the URL to your clipboard.

   > **Note:** Currently, the URL copied to your clipboard begins with http://, not https://. To use the URL, you must manually change this to https://.

   f. Paste the URL into a text editor to identify the endpoint host. For example, the URL is similar to the following:

   ```
   http://dfdj6kgx.cde-2cdxw5x5.apps.ecs-demo.example.com/dex/api/v1
   ```

   The endpoint host is dfdj6kgx.cde-2cdxw5x5.apps.ecs-demo.example.com.

6. On the ECS or HDFS gateway host you selected previously, initialize the virtual cluster using the cde-utils.sh
script. You can either generate and use a self-signed certificate, or provide a signed certificate and private key.

Generate a self-signed certificate

```
./cde-utils.sh init-virtual-cluster -h <endpoint_host> -a
```

For example, using the previous example URL, the endpoint host is dfdj6kgx.cde-2cdxw5x5.apps.ecs-demo.exam
ple.com:

```
./cde-utils.sh init-virtual-cluster -h dfdj6kgx.cde-2cdxw5x5.apps.ecs-de
mo.example.com -a
```

Use a signed certificate and private key

Make sure that the certificate is a wildcard certificate for the cluster endpoint. For example, *.cde-2cdxw5x5.apps.
ecs-demo.example.com

```
./cde-utils.sh init-virtual-cluster -h <endpoint_host> -c /path/to/cert -
k /path/to/keyfile
```

For example, using the previous example URL, the endpoint host is dfdj6kgx.cde-2cdxw5x5.apps.ecs-demo.exam
ple.com:

```
./cde-utils.sh init-virtual-cluster -h dfdj6kgx.cde-2cdxw5x5.apps.ecs-de
mo.example.com  -c /tmp/cde-pvc.crt -k /tmp/cde-pvc.key
```

> **Note:** You can use the password protected private key to initialize the virtual cluster. When prompted,
> enter the password. Currently, the password protected private keys are supported with the RSA and EC
> algorithms only.

You must perform this procedure for each virtual cluster you create.

Managing virtual clusters

You can view configuration, metrics, and logs of existing virtual clusters. You can use the Edit option to make the
configuration changes dynamically. You can also view jobs associated with a cluster, clone a cluster, and delete a
cluster.

To view and edit cluster details:

1. In the Cloudera Data Platform (CDP) console, click the Data Engineering tile. The CDE Home page displays.
2. Click Administration in the left navigation menu. The Administration page displays.
3. In the Services column, select the environment containing the virtual cluster you want to manage.
4. In the Virtual Clusters column on the right, click the Cluster Details icon for the virtual cluster you want to
manage.
5. On the Configuration tab, you can view details about the cluster, including the cluster name and CDP
environment.

6. Optional: Click the Edit option to make the configuration changes dynamically which may take a few minutes to update.

> ⚠️ **Important:** When you edit the log retention policy configuration, you must restart the runtime-api-server pod using the `kubectl rollout restart deployment/<deployment-name> -n <namespace>` command to apply the changes.
>
> For example:
>
> ```
> kubectl rollout restart deployment/dex-app-fww6lrgm-api -n dex-app-f
> ww6lrgm
> ```

The virtual cluster Overview page provides details about the virtual cluster, and includes links to several resources, such as the API documentation, and the Jobs API URL. You can switch between the following tabs to view additional information:

Configuration

> The Configuration tab lists details about the cluster, including the cluster name, CDP environment, and the CPU and memory capacity. You can modify the virtual cluster's CPU and memory capacity dynamically.

Charts

> The Charts tab displays the charts related to CPU Requests, Memory Requests, Jobs, and Pods.

Logs

> The Logs tab displays the latest log entries for the virtual cluster.

Deleting virtual clusters

If you no longer need a particular virtual cluster, you can delete it. Deleting a virtual cluster instantly deletes all metadata associated with the cluster, including job configurations and log files. Do not delete a virtual cluster unless you are certain you no longer need access to any of the applications, jobs, or log files. Job run history is maintained separately, and is not deleted even if you delete a virtual cluster.

Before you begin

Procedure

1. In the Cloudera Data Platform (CDP) console, click the Data Engineering tile. The CDE Home page displays.

2. Click Administration in the left navigation menu. The Administration page displays.

3. In the Services column, select the environment containing the virtual cluster you want to delete.

4. In the Virtual Clusters column for the selected environment, click ⋮ at the top right of the virtual cluster that you want to delete, and then click Delete.

> ⚠️ **Warning:** Deleting a virtual cluster deletes all metadata associated with the cluster, including job configurations and log files. Do not do this unless you are certain that you no longer need any of these.

5. Confirm that you want to delete the cluster by entering the cluster name and then clicking Delete.