

Cloudera Operational Database ..

## Release Notes

Date published: 2020-08-14

Date modified: 2024-12-18

# CLOUDERA

<https://docs.cloudera.com/>

# Legal Notice

© Cloudera Inc. 2025. All rights reserved.

The documentation is and contains Cloudera proprietary information protected by copyright and other intellectual property rights. No license under copyright or any other intellectual property right is granted herein.

Unless otherwise noted, scripts and sample code are licensed under the Apache License, Version 2.0.

Copyright information for Cloudera software may be found within the documentation accompanying each component in a particular release.

Cloudera software includes software from various open source or other third party projects, and may be released under the Apache Software License 2.0 (“ASLv2”), the Affero General Public License version 3 (AGPLv3), or other license terms. Other software included may be released under the terms of alternative open source licenses. Please review the license and notice files accompanying the software for additional licensing information.

Please visit the Cloudera software product page for more information on Cloudera software. For more information on Cloudera support services, please visit either the Support or Sales page. Feel free to contact us directly to discuss your specific needs.

Cloudera reserves the right to change any products at any time, and without notice. Cloudera assumes no responsibility nor liability arising from the use of products, except as expressly agreed to in writing by Cloudera.

Cloudera, Cloudera Altus, HUE, Impala, Cloudera Impala, and other Cloudera marks are registered or unregistered trademarks in the United States and other countries. All other trademarks are the property of their respective owners.

Disclaimer: EXCEPT AS EXPRESSLY PROVIDED IN A WRITTEN AGREEMENT WITH CLOUDERA, CLOUDERA DOES NOT MAKE NOR GIVE ANY REPRESENTATION, WARRANTY, NOR COVENANT OF ANY KIND, WHETHER EXPRESS OR IMPLIED, IN CONNECTION WITH CLOUDERA TECHNOLOGY OR RELATED SUPPORT PROVIDED IN CONNECTION THEREWITH. CLOUDERA DOES NOT WARRANT THAT CLOUDERA PRODUCTS NOR SOFTWARE WILL OPERATE UNINTERRUPTED NOR THAT IT WILL BE FREE FROM DEFECTS NOR ERRORS, THAT IT WILL PROTECT YOUR DATA FROM LOSS, CORRUPTION NOR UNAVAILABILITY, NOR THAT IT WILL MEET ALL OF CUSTOMER’S BUSINESS REQUIREMENTS. WITHOUT LIMITING THE FOREGOING, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, CLOUDERA EXPRESSLY DISCLAIMS ANY AND ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, QUALITY, NON-INFRINGEMENT, TITLE, AND FITNESS FOR A PARTICULAR PURPOSE AND ANY REPRESENTATION, WARRANTY, OR COVENANT BASED ON COURSE OF DEALING OR USAGE IN TRADE.

# Contents

<b>What's New.....</b>	<b>4</b>
January 15, 2025.....	4
December 11, 2024.....	4
November 5, 2024.....	5
September 3, 2024.....	6
May 30, 2024.....	8
April 28, 2024.....	9
March 28, 2024.....	10
February 28, 2024.....	10
February 7, 2024.....	10
Older releases.....	11
December 18, 2023.....	11
October 26, 2023.....	12
September 28, 2023.....	13
August 30, 2023.....	13
August 10, 2023.....	14
June 19, 2023.....	15
May 31, 2023.....	16
May 10, 2023.....	16
March 10, 2023.....	16
January 10, 2023.....	17
December 10, 2022.....	17
November 10, 2022.....	18
October 10, 2022.....	18
August 10, 2022.....	19
June 10, 2022.....	19
April 10, 2022.....	20
March 10, 2022.....	20
February 10, 2022.....	20
December 10, 2021.....	21
November 10, 2021.....	21
October 10, 2021.....	21
September 10, 2021.....	22
August 10, 2021.....	22
July 10, 2021.....	23
May 10, 2021.....	23
February 10, 2021.....	24
November 10, 2020.....	24
 <b>Fixed issues.....</b>	 <b>24</b>
 <b>Known issues.....</b>	 <b>25</b>

## What's New

This section lists major features and updates for the Cloudera Operational Database experience in CDP Public Cloud.

### January 15, 2025

The Cloudera Operational Database 1.48 version supports the SELinux enforcement and enhancement to the database user management.

#### Cloudera Operational Database supports Security-Enhanced Linux (SELinux) enforcement

Cloudera Operational Database supports creating a database with SELinux enforcement using the Cloudera Operational Database CLI. You must have the CDP\_SECURITY\_ENFORCING\_SELINUX entitlement to be able to use the SELinux support. Please contact Cloudera support if you do not have this entitlement.

The SELinux allows you to set access control through policies. You can set the SELinux mode while creating a new operational database. You can define the SELinux mode using the seLinux parameter in the create-database command. The supported SELinux modes are:

- ENFORCING: Enables SELinux in enforced mode, actively applying security policies.
- PERMISSIVE (default): Sets SELinux to permissive mode, logging any security violations without enforcing policies.

If you do not define the seLinux parameter, by default, the PERMISSIVE mode is applied.

The following example shows usage of the seLinux parameter.

```
opdb create-database --environment-name [***ENVIRONMENT_NAME***] --database-name [***DATABASE_NAME***] --security-request '{"seLinux": string}'
```

```
opdb create-database --environment-name cod-7218-microl --database-name test DB --security-request '{"seLinux": "ENFORCING"}'
```

```
opdb create-database --environment-name cod-7218-microl --database-name test DB --security-request '{"seLinux": "PERMISSIVE"}'
```

For more information, see *Cloudera CLI documentation* and *Setting SELinux Mode*.

#### Assign OAdmin role at the database level

Cloudera Operational Database supports setting a user as an OAdmin at the database level. In earlier versions of the Cloudera Operational Database, you could set the OAdmin at the environment level only, however, for better usability and enhanced security, now you can set it at the database level too.

#### Related Information

[Cloudera CLI documentation](#)

[Setting SELinux Mode](#)

### December 11, 2024

The Cloudera Operational Database 1.47 version supports enhancements to the CDP CLI and adding a new storage type while creating a new operational database.

### Cloudera Operational Database supports JDK17

Cloudera Operational Database only supports creating a database with Cloudera Runtime 7.3.1 using JDK17. If you use an earlier runtime version, only JDK8 and JDK11 are supported for database creation.

### New CLI commands to list and upload certificates

Cloudera Operational Database supports two new CLI commands, `list-certificates` and `upload-certificate`.

In an Auto-TLS setup, Cloudera Manager maintains a global certificate trust store across the cluster to ensure a mutual trust relationship between cluster nodes in secure TLS connections.

You can now upload custom certificates into the global certificate store and distribute them across all nodes, enabling secure COD connections from your infrastructure without changing the existing PKI infrastructure, certificates, or Root CA.

The feature is designed to support mTLS authentication from outside of COD's network, but it could also be useful for TLS connections from COD to other networks in general. The command details are as follows.

- `list-certificates`: This command lists SHA-1 fingerprints of certificates listed in the Global Trust Store.

Following is an example,

```
cdp opdb list-certificates --environment <environment_name> --database <database_name>
```

- `upload-certificate`: This command uploads a single, PEM-encoded certificate to the Global Trust Store and refreshes all the nodes in the cluster.

Following is an example,

```
cdp opdb upload-certificate --environment <environment_name> --database <database_name> --certificate <custom_certificate_in_PEM_format>
```

For more information, see *CDP CLI documentation*.

### New storage type support during database creation

Cloudera Operational Database UI supports a new storage type, Cloud Storage with Caching and Data Tiering while creating an operational database. This storage type is equivalent to a cloud storage that supports time-based priority caching, where data within a specified time range is given a higher priority.

You must have the `COD_DATATIERING` entitlement to be able to use this storage type.

For more information, see *Creating a database using COD*.

### Related Information

[CDP CLI documentation](#)

[Creating a database using COD](#)

## November 5, 2024

The Cloudera Operational Database 1.46 version supports updates to the supported instance types and enhancements to the Operational Database UI.

### A new CLI command to get the list of supported instance types

Cloudera Operational Database adds a new CLI command, `list-supported-instance-types`, allowing you to list the supported instance types for a combination of cloud platform, scale type, and storage type. You can filter using the instance group and architecture as well.

The `--custom-instance-types` option in the `create-database` command allows you to pick and choose instance types for the COD. However, it is essential that these instance types are included in the COD allowlist. The `list-supported-instance-types` command is useful to fetch the list of supported instance types.

Following is an example of the command.

```
cdp opdb list-supported-instance-types --cloud-platform AZURE --storage-type
  CLOUD_WITH_EPHEMERAL --scale-type MICRO --instance-group WORKER --architect
  ure X86_64
```

For more information, see *CDP CLI documentation*.

### Enhancements to the update-database command

A new CLI option `--switch-instance-type` is added to the `update-database` command to switch the instance types to custom instance types for the COD clusters after the database creation. You must have the `COD_USE_CUSTOM_INSTANCE_TYPES` entitlement to use this option.

If you plan to scale up the cluster vertically or change the existing instance type to any custom instance type, you can use the `--switch-instance-type` option. However, the custom instance types must be included in the allowlist by the Cloudera Operational Database. Use the `list-supported-instance-types` command to fetch the list of supported instance types.

Following is an example of the usage.

```
cdp opdb update-database --environment-name <env_name> --database-name <data
  base_name> --switch-instance-type instanceGroup=INSTANCEGROUP,instanceType=i
  nstancetype
```

For more information, see *CDP CLI documentation*.

### Enhancements to the Cloudera Operational Database UI

Cloudera Operational Database UI provides two new UI options `Configure Database` and `Configure Edge Nodes` to configure the worker and edge node properties in your existing operational database.

You can access these menu items using the `Actions` menu item inside the database details page.

For more information, see *Configuring a database*.

### GA support for the HBase REST server scaling

The feature to scale up the HBase REST server using the Apache HBase REST API is no longer a technical preview and is now generally available (GA).

For more information, see *HBase REST server scaling in COD*.

### Related Information

[CDP CLI documentation](#)

[Configuring a database](#)

[HBase REST server scaling in COD](#)

## September 3, 2024

The Cloudera Operational Database 1.45 version supports updates to the HDFS instance types and enhancements to the Operational Database UI.

### Enhancements to the create-database command

- The CDP CLI adds a new option `--custom-instance-types` to the `create-database` command. Using this option, you can define custom instance types; however, the instance types must be included in the allowlist by Operational Database.

Currently, the allowlist is not accessible, however, the following are the new custom instance types supported by Operational Database.

AWS, HEAVY, HDFS:

```
Worker: m6i.4xlarge, m7i.4xlarge
Master: m6i.8xlarge, m7i.8xlarge
Compute/edge/leader: m6i.2xlarge, m7i.2xlarge
Gateway: r6i.8xlarge, r7i.8xlarge
```

AWS, LIGHT, HDFS:

```
Worker: m6i.4xlarge, m7i.4xlarge
Compute/edge/leader/gateway: m6i.2xlarge, m7i.2xlarge
Master: m6i.4xlarge, m7i.4xlarge
```

The following is a sample output of the `create-database` command highlighting the usage of the `--custom-instance-types` option.

```
cdp opdb create-database
--environment-name cod_env
--database-name cod_db
--custom-instance-types masterType=m7i.4xlarge,workerType=m7i.4xlarge,leaderType=m7i.2xlarge,gatewayType=m7i.2xlarge
--storage-type=HDFS
--scale-type LIGHT
```

- The `--storage-type` option is now optional. If you do not define the `--storage-type` option, Operational Database considers the default storage type.

The default storage type is blob storage. If the ephemeral storage is enabled, Operational Database considers the storage type as `CLOUD_WITH_EPHEMERAL`; otherwise, `CLOUD` is considered.

For more information, see *CDP CLI documentation*.

### Updates to the HDFS clusters on AWS environments to add support for m6i and m7i instance types

When you create an operational database with HDFS storage type in an AWS environment, Operational Database on HDFS clusters now also supports m6i and m7i instances for the applicable nodes. The Operational Database clusters with HDFS storage type are upgraded to enhance the operational database's performance and usability.

The following are the new custom instance types supported by Operational Database.

AWS, HEAVY, HDFS:

```
Worker: m6i.4xlarge, m7i.4xlarge
Master: m6i.8xlarge, m7i.8xlarge
Compute/edge/leader: m6i.2xlarge, m7i.2xlarge
Gateway: r6i.8xlarge, r7i.8xlarge
```

AWS, LIGHT, HDFS:

```
Worker: m6i.4xlarge, m7i.4xlarge
Compute/edge/leader/gateway: m6i.2xlarge, m7i.2xlarge
Master: m6i.4xlarge, m7i.4xlarge
```

## Enhancements to the Cloudera Operational Database UI

The Cloudera Operational Database UI is updated for better usability and performance. The following are the key enhancements.

- The cloud form factor on which the database is deployed is shown as a logo under the Environment column.
- The Version column is renamed to the Runtime Version in the Databases screen.
- A new column Node Count is added to the Databases screen.
- The Date Created column name is changed to Uptime in the Databases screen.
- The SQL Editor is renamed HUE and displayed as a link under the new column Apps.
- A new action menu item Collect diagnostic bundle is added to the Databases *DATABASE\_NAME* Actions . For more information, see *Collecting operational database diagnostic bundle*.

### Related Information

[CDP CLI documentation](#)

[Collecting operational database diagnostic bundle](#)

## May 30, 2024

Cloudera Operational Database (COD) 1.42 version supports HBase REST server scaling and CDP CLI enhancements.

### The HBase REST server scaling for better performance [Technical Preview]

You can scale up the HBase REST server using the Apache HBase REST API, for better connectivity to COD. You need a minimum of two Gateway nodes to utilize this functionality. The required number of Gateway nodes can be specified using the `--gateway-nodes-count` option in the `create-database` command using CDP CLI.

This feature is under technical preview. To use this feature, you must have the `COD_RESTWORKERS` entitlement enabled in your CDP environment.

Following is a sample command.

```
cdp opdb create-database --environment-name ENV_NAME --database-name DATABASE_NAME --gateway-nodes-count INTEGER
```

For more information, see *Scaling the HBase REST server in COD*.

### Enhancements to the describe-database command

In CDP CLI, the output of the `describe-database` command shows the JDK version of the COD cluster if the cluster was created using a specific JDK version; otherwise, the output shows the JDK version as "Not Available".

The following is a sample output of the `describe-database` command that shows the Java version used to create the cluster.

```
"dbEdgeNodeCount": 0 ,
"scaleType": "MICRO" ,
"type": "COD" ,
"computeNodesCount": 0 ,
"totalComputeNodesCount": 0 ,
"isJwtEnabled": true ,
"cloudPlatform": "AWS" ,
"javaVersion": "11"
```

For more information, see *CDP CLI documentation*.

### Related Information

[Scaling the HBase REST server in COD](#)

[CDP CLI documentation](#)



## April 28, 2024

Cloudera Operational Database (COD) 1.41 version supports CDP CLI changes and upgrade to higher instance types for HDFS storage type and Azure deployments.

### COD has updated the HDFS instance type to 16 core instances

When you create an operational database with HDFS storage type, the COD clusters now use 16 core instances on AWS, Azure, and GCP environments for worker nodes. The COD clusters with HDFS storage type are upgraded to enhance the performance and usability of the COD.

The new worker instances for HDFS storage type are as follows:

- AWS: m5.4xlarge
- Azure: Standard\_D16\_v3
- GCP: n2-standard-16

### COD supports configuring root volume size for available instances in a COD cluster

In CDP CLI, while creating an operational database, you can set the default root volume size with the `--root-volume-size` (integer) option in GiB for all the instances in the cluster.

Following is a sample command.

```
cdp opdb create-database --environment-name test-env --database-name test-db
--root-volume-size 300
```

For more information, see *CDP CLI documentation*.

### COD has updated the instance type for Azure deployments

When you create and deploy an operational database in an Azure environment, by default, COD clusters now use Standard\_D8s\_v3 instance type instead of Standard\_D8\_v3. The instance type is upgraded to support encryption at the host level.

If you want to retain the Standard\_D8\_v3 instance type, you must have the `COD_USE_DV3_INSTANCE_TYPE` entitlement on your account.

### COD supports instance group encryption in AWS environments

In CDP CLI, while creating an operational database, you can specify the encryption key to encrypt the volume for instance groups using the `--volume-encryptions` (array) option. You can specify this option only in AWS environments.

Following is a sample command.

```
opdb create-database --environment-name <environment-name> --database-name <
database-name> --disable-external-db --scale-type MICRO --attached-storage-f
or-workers '{"volumeCount":1,"volumeType":"SSD","volumeSize":100}' --endpoi
nt-url http://localhost:8988 \
--volume-encryptions '[
  {
    "encryptionKey": "<aws-key-arn>",
    "instanceGroup": "GATEWAY"
  }
]'
```

Shorthand syntax: `encryptionKey=string.instanceGroup=string ...` (separate items with spaces)

JSON syntax:

```
[
    {
        "instanceGroup": "WORKER" | "LEADER" | "MASTER" | "GATEWAY" | "STRONG
        META" | "EDGE"
    }
    ...
]
```

For more information, see *CDP CLI documentation*.

#### Related Information

[CDP CLI documentation](#)

## March 28, 2024

Cloudera Operational Database (COD) 1.40 version provides enhancements to the create database UI wizard.

### COD supports enhancements to the Create Database UI wizard

When you create an operational database using the COD UI, the Create Database option provides enhancements to the database creation workflow. In addition to the enhanced look and feel for better usability, the workflow also provides two additional steps called, Settings and Review. In the Settings step, you can view the default settings for your database and can also modify them. In the Review step, you can view, and finalize your changes, and also obtain a complete CLI command, in case you want to create the database using CDP CLI.

For more information, see *Creating a database using COD*.

#### Related Information

[Creating a database using COD](#)

[CDP CLI Beta](#)

## February 28, 2024

Cloudera Operational Database (COD) 1.39 version removes a CDP CLI command and provides support for GP3 for attached storage.

### COD has removed the CDP CLI command, `disengage-auto-admin`

COD has removed the support for `disengage-auto-admin` command, which allowed users to disable the autonomous functions of the database and use the underlying DataHub cluster instead.

### COD supports GP3 for attached storage disks

COD now supports GP3 (SSD) volume types for attached storage. GP3 volumes allow you to increase performance (independently provisioning IOPS and throughput) without increasing storage size. GP3 volumes deliver similar performance as similar GP2 volumes at a lower cost. GP3 is now the default attached storage type for COD instances that previously used GP2 storage.

## February 7, 2024

Cloudera Operational Database (COD) 1.38 version introduces a new CDP CLI command.

### COD has introduced a new CDP CLI command `prepare-upgrade-database`

The new CDP CLI command `prepare-upgrade-database` is used during the preparation phase of the runtime upgrades. This command performs all validations and downloads all required parcels for the upgrade operation. Following is a sample command.

```
cdp opdb prepare-upgrade-database --environment <ENVIRONMENT-NAME> --database <DATABASE-NAME> [--runtime <RUNTIME-VERSION>] [--imageId <IMAGEID>]
```

You can either use the runtime version `--runtime` or the image ID `--imageId` of the cluster while using this command.

For more information, see *CDP CLI Beta*.

### Related Information

[CDP CLI Beta](#)

## Older releases

Overview of new features, enhancements, and changed behavior introduced in earlier releases of Cloudera Operational Database.

### December 18, 2023

Cloudera Operational Database (COD) 1.37 version supports modifications to the entitlements and Multi-AZ deployment on an Azure environment.

#### COD has enabled the `COD_ON_GCS` entitlement

You can deploy COD on a Google Cloud Platform (GCP) by using Google Cloud Storage (GCS) similar to what is available for Amazon Web Services (AWS) S3 storage and Microsoft Azure blob storage. Now, COD has enabled the `COD_ON_GCS` entitlement, by default, for such a deployment.

#### COD has removed the `COD_EDGE_NODE` entitlement

COD has removed the `COD_EDGE_NODE` entitlement now because it is not needed anymore. COD edge node functionality is enabled for all COD customers now.

#### COD has removed the `COD_STOREFILE_TRACKING` entitlement

COD has removed the `COD_STOREFILE_TRACKING` entitlement because it is not needed anymore. The Store File Tracking (SFT) functionality is enabled on all new COD clusters created with cloud storage.

#### COD has removed the `OPDB_USE_EPHEMERAL_STORAGE` entitlement

COD has removed the `OPDB_USE_EPHEMERAL_STORAGE` entitlement because it is not needed anymore. The use of COD on a cloud storage with ephemeral cache is enabled without an entitlement depending on the cluster creation parameters.

#### COD supports Multiple Availability Zones (Multi-AZ) on Azure [Technical Preview]

COD ensures high availability and fault tolerance using Multi-AZ deployments. A Multi-AZ deployment means that compute infrastructure for HBase's master and region servers are distributed across multiple AZs ensuring that when a single availability zone has an outage, only a portion of Region Servers is impacted and clients automatically switch over to the remaining servers in the available AZs.

Multi-AZ for COD is now supported on Microsoft Azure environments as a technical preview and is considered under development. For more information, see *Multi-AZ deployment on COD*.

### **COD supports a new instance type I4i for Cloud With Ephemeral Storage type databases on AWS environments**

When you create a new operational database with Cloud With Ephemeral Storage as the storage type on an AWS environment, COD creates the database with an I4i instance type for the worker nodes.

### **COD supports fast autoscaling for higher computing requirements**

When you create a new operational database using CDP CLI, you can enable fast autoscaling by defining the required parameters using the `--auto-scaling-parameters` option. COD now supports a new instance group called, Compute. The nodes under this instance group are automatically scaled up or scaled down based on the CPU utilization and RPC latency.

To use fast autoscaling, you must have the `COD_USE_COMPUTE_ONLY_NODES` entitlement.

Following is a sample command.

```
cdp opdb create-database --environment-name <ENV_NAME> --database-name <DB_NAME>
--auto-scaling-parameters '{"minComputeNodesForDatabase":<min_compute_nodes>, "maxComputeNodesForDatabase": <max_compute_nodes>}'
```

For more information, see *The fast autoscaling in COD*.

#### **Related Information**

[Multi-AZ deployment on COD](#)

[Fast autoscaling in COD](#)

## **October 26, 2023**

Cloudera Operational Database (COD) 1.36 version supports an UI enhancement and the enabling of multiple entitlements.

### **COD has enabled the `OPDB_USE_EPHEMERAL_STORAGE` entitlement**

COD supports large ephemeral block cache while deploying on cloud storage. The entitlement `OPDB_USE_EPHEMERAL_STORAGE` is enabled by default while using a large ephemeral block cache on any cloud storage.

### **COD introduces a new storage type UI option while creating an operational database**

On the COD UI, a new storage type option Cloud With Ephemeral Storage is added. This option is equivalent to using the `--storage-type CLOUD_WITH_EPHEMERAL` option on CDP CLI while creating an operational database.

For more information, see *Creating a database using COD* and *CDP CLI Beta*.

### **COD has enabled the `COD_EDGE_NODE` entitlement**

Earlier, you were required to have the `COD_EDGE_NODE` entitlement to create an edge node on your COD cluster. Now the entitlement is enabled by default.

### **COD has enabled the `COD_STOREFILE_TRACKING` entitlement**

Earlier, you were required to have the `COD_STOREFILE_TRACKING` entitlement to use the Store File Tracking (SFT) on your COD cluster. Now the entitlement is enabled by default.

#### **Related Information**

[Creating a database using COD](#)

[CDP CLI Beta](#)

## September 28, 2023

Cloudera Operational Database (COD) 1.35 version supports rolling operating system upgrade and enhancements to the CDP CLI.

### COD supports rolling operating system upgrades of a COD cluster

COD using *HDFS* and *Object Store without Ephemeral Storage*, now supports upgrading the operating system version of the database using the rolling restart mode. This ensures continuous service availability during an upgrade operation.

In CDP CLI, you need to use the `--os-upgrade-only` option along with the `--rolling-upgrade` option in the `upgrade-database` command.

Following is a sample command.

```
cdp opdb upgrade-database --environment <ENVIRONMENT-NAME> --database <DATABASE-NAME> [--runtime <RUNTIME-VERSION> | --image <IMAGEID>] --os-upgrade-only --rolling-upgrade
```

For more information, see *Performing a Cloudera operating system upgrade*.

### COD introduces a new CDP CLI option `--storage-type` in the `create-database` command

In CDP CLI, a new option `--storage-type` in the `create-database` command is introduced that replaces the `[--use-hdfs | --no-use-hdfs]` and `[--disable-ephemeral-storage | --no-disable-ephemeral-storage]` options.

The `--storage-type` option supports these values, `CLOUD_WITH_EPHEMERAL`, `CLOUD`, and `HDFS`.

Following is a sample command.

```
cdp opdb create-database --environment <ENVIRONMENT-NAME> --database <DATABASE-NAME> --storage-type CLOUD
```

For more information, see CDP CLI Beta.

### COD drops support of the Cloudera runtime versions CDP Runtime 7.2.12 and earlier

COD has stopped supporting the Cloudera runtime versions CDP Runtime 7.2.12 and earlier because they have reached the end of life.

#### Related Information

[CDP CLI Beta](#)

[Performing a Cloudera operating system upgrade](#)

## August 30, 2023

Cloudera Operational Database (COD) 1.34 version supports different JDK versions during COD creation and deploying COD on GCS.

### COD supports creating an operational database using JDK8 and JDK11

COD now added a new CLI option, `--java-version` which can be used to configure a major Java version on your COD cluster. The new CLI option can be used along with the `create-database` command to specify the Java version. The supported Java versions are JDK8 and JDK11. In case the parameter is not specified, JDK8 is used. Following is a sample command.

```
cdp opdb create-database --environment-name <ENVIRONMENT_NAME> --database-name <DATABASE_NAME> --java-version <VALUE>
```

```
cdp opdb create-database --environment-name cod7215 --database-name testenv --java-version 11
```

For more information, see *CDP CLI beta*.

### COD is available as a Technical Preview feature on Google Cloud Storage (GCS)

COD on Google Cloud Platform (GCP) can now be deployed by using Google Cloud Storage (GCS) easily, similar to what is available for Amazon Web Services (AWS) S3 storage and Microsoft Azure blob storage. The use of GCS for such a setup requires the COD\_ON\_GCS entitlement.

COD also now supports a large ephemeral block cache while deploying on GCP. The use of ephemeral storage along with any cloud storage still requires the OPDB\_USE\_EPHEMERAL\_STORAGE entitlement.

### COD has removed the COD\_ON\_GCP entitlement

COD\_ON\_GCP entitlement has been removed from COD because it is not needed anymore. From this version onwards, customers can create COD clusters on Google Cloud Platform (GCP) without it.

### Related Information

[CDP CLI Beta](#)

## August 10, 2023

Cloudera Operational Database (COD) 1.33 version provides enhancements to the CDP CLI as well as on COD UI.

### COD drops support of the Cloudera runtime versions CDP Runtime 7.2.8 and CDP Runtime 7.2.9

COD has stopped supporting the Cloudera runtime versions CDP Runtime 7.2.8 and CDP Runtime 7.2.9 because they have reached the end of life.

### COD supports faster rolling restarts on COD clusters

The default value of Cloudera Manager HBase Configuration Region Mover Threads is changed to 30. This speeds up the rolling restart functionality for HBase.

For more information see *Rolling Restart*.

### COD supports rolling runtime upgrades of a COD cluster

COD now supports upgrading the Cloudera Runtime version of the database using the rolling restart mode. This ensures continuous service availability during an upgrade operation. A new CLI parameter `--rolling-upgrade | --no-rolling-upgrade` is added to the `upgrade-database` command. Following is a sample command:

```
cdp opdb upgrade-database --environment <environment-name> --database <database-name> --runtime <runtime-version> [--rolling-upgrade | --no-rolling-upgrade]
```

For more information, see *Performing a rolling Cloudera Runtime upgrade*.

### COD provides enhancements to the `--scale-type` CDP CLI option in the `create-database` command

In CDP CLI, the `--scale-type` option now supports all three options `--scale-type (string) <MICRO, LIGHT, HEAVY>` for both the `--master-node-type` and `--gateway-node-type`.

- `--scale-type LIGHT` (`--master-node-type LITE`, `--gateway-node-type LITE`)
- `--scale-type HEAVY` (`--master-node-type HEAVY`, `--gateway-node-type HEAVY`)

If the `--scale-type` option is not defined, by default `--scale-type LIGHT` is considered for both the `--master-node-type` and `--gateway-node-type`. However, you can overwrite the `--scale-type` for a `--gateway-node-type` using the `--gateway-node-type <value>` option.

For more information, see *CDP CLI Beta*.

### COD supports enabling a consolidated view of COD metrics using Grafana dashboards

In CDP CLI, the `create-database` command now provides a new option `--enable-grafana` which allows you to enable the Grafana URL under the `GRAFANA DASHBOARD` option inside your COD database. When you click on the Grafana URL, it takes you to the Grafana dashboard which provides a consolidated view of the COD metrics.

Following is an example of the `create-database` command.

```
cdp opdb create-database --environment <environment_name> --database <database_name> --enable-grafana
```

For more information, see *Monitoring metrics in COD with Grafana*.

#### Related Information

[Rolling Restart](#)

[Performing a Cloudera Runtime upgrade](#)

[CDP CLI Beta](#)

[Monitoring metrics in COD with Grafana](#)

## June 19, 2023

Cloudera Operational Database (COD) 1.32 version provides enhancements to the CDP CLI as well as on COD UI.

### COD provides UI enhancements to the Scale option on the database creation page

On the COD UI, when you create an operational database the Medium Duty is renamed to Heavy Duty under Create Database Scale . This ensures that the options on COD UI and CDP CLI Beta are symmetrical.

For more information, see *Creating a database using COD*.

### COD provides enhancements to the CDP CLI option `--scale-type <HEAVY>`

In CDP CLI, when you select the `--scale-type` option as `HEAVY`, COD allocates larger SSD storage (for example, `gp2` on AWS, `StandardSSD_LRS` on Azure, or `pd-ssd` on GCP) for both master and leader node types. This ensures the higher loads on Zookeeper and provides a better performance for COD.

For more information, see *CDP CLI Beta*.

### COD supports enabling custom recipes using CDP CLI Beta

You can now define custom recipes while creating an operational database using the `--recipes` option in CDP CLI Beta. You can register pre-created recipes during database creation based on the instance groups in your database. The recipes are executed automatically for the specified nodes based on the recipes' type.

Use the following example command to define custom recipes for your operational database.

```
cdp opdb create-database --environment-name <ENVIRONMENT_NAME> --database-name <DATABASE_NAME> --recipe names=<rec1,rec2>,instanceGroup=<MASTER> names=<rec2>,instanceGroup=WORKER names=<rec3,rec4>,instanceGroup=<GATEWAY>
```

To know more about the recipes, see *Recipes*.

To know more about the `--recipes` option, see *CDP CLI Beta*.

#### Related Information

[Creating a database using COD](#)

[CDP CLI Beta](#)

[Recipes](#)

## May 31, 2023

Cloudera Operational Database (COD) 1.31 version provides enhancements to the CDP CLI options.

### COD provides enhancements to the `--scale-type` CDP CLI option

In CDP CLI, the `--scale-type` option now supports all the three options `--scale-type (string) <MICRO, LIGHT, HEAVY>`. COD has added the support for additional parameters, `LIGHT` and `HEAVY`.

- `--scale-type LIGHT` (master-node-type `LITE`)
- `--scale-type HEAVY` (master-node-type `HEAVY`)

Additionally, COD has removed the `--master-node-type (string) <LITE,HEAVY>` CDP CLI option because this option is not needed anymore. For more information, see *CDP CLI Beta*.

### Related Information

[CDP CLI Beta](#)

## May 10, 2023

Cloudera Operational Database (COD) 1.30 version supports scaling up the COD clusters vertically and also provides an UI option to create smaller COD clusters.

### COD supports scaling up the clusters vertically

COD now allows you to vertically scale up the COD clusters from a Light Duty to a Medium Duty instance type. You can upgrade the instance type of a COD cluster that belongs to a Master or Gateway node.

To know more about the vertical scaling, see *Scaling COD instances vertically*.

### COD UI supports creating a smaller cluster using a predefined Data Lake template

COD now allows you to create a smaller cluster with one Gateway node and one Worker node using a new scale type Micro Duty while creating an operational database through COD UI. The Micro database is a two node cluster in which the Gateway node processes the activities of the Master and Leader nodes, thereby removing the need of these nodes. You can use a Micro database for testing and development purposes.

For more information, see *Creating a database using COD*.

### Related Information

[Scaling COD instances vertically](#)

[Creating a database using COD](#)

## March 10, 2023

Cloudera Operational Database (COD) 1.28 version provides UI enhancements that include storage type selection when you create a database and JWT configurations to connect to your HBase client.

### COD UI allows storage type selection when creating an operational database

COD UI now allows you to select the storage type when creating an operational database. You can either select Cloud Storage or HDFS.

Earlier if you had to use the HDFS as the storage type, you were required to use the `--use-hdfs` option on COD CLI while creating the operational database. Now COD UI is enhanced to let you select the storage type when you create an operational database.

For more information, see *Creating a database using COD*.



### COD UI provides the JWT configuration details to connect to your HBase client

Now you can find the JWT configuration details On the COD UI. Click on a database and go to [Connect HBase Client Tarball JWT Configuration](#) . You can refer to these configurations to set up a connection to HBase with a JWT token and build your own truststore JKS file.

For more information, see [Configuring JWT authentication for HBase client](#).

#### Related Information

[Creating a database using COD](#)

[Configuring JWT authentication for HBase client](#)

## January 10, 2023

Cloudera Operational Database (COD) 1.27 version supports JWT authentication, provides Data Lake templates while creating a database, and a CLI option to enable HBase region canaries.

### COD supports configuring JWT authentication for your HBase clients

COD now allows you to configure JWT (JSON Web Token)-based authentication for your HBase clients, which uses a unique identifier and is a standard way of securely transmitting signed information between two parties. To know more about the JWT authentication, see [Configuring JWT authentication for HBase client](#).

### COD supports creating an operational database using a predefined Data Lake template

When you create an operational database, you can now define the structure of your database based on a predefined Data Lake template. The template defines the number of gateway, master, and worker nodes to be added while creating a database.

You can select a template and accordingly the nodes are added into the COD cluster after the database is successfully created.

To know more about this, see [Creating a database using COD](#).

### COD provides a CLI option to enable HBase region canaries

COD now provides a CLI option, `--enable-region-canary` to enable the HBase region canaries while creating an operational database.

Use the following command to enable the HBase region canaries.

```
cdp opdb create-database --environment-name ENVIRONMENT_NAME --database-name DATABASE_NAME --enable-region-canary
```

- `hbase_region_health_canary_enabled`
- `hbase_region_health_canary_slow_run_alert_enabled`
- `hbase_canary_alert_unhealthy_region_percent_threshold`

For more information, see [Enabling HBase region canary](#).

#### Related Information

[Configuring JWT authentication for HBase client](#)

[Creating a database using COD](#)

[Enabling HBase region canary](#)

## December 10, 2022

Cloudera Operational Database (COD) 1.26 version supports managing the edge nodes using COD CLI and disabling the Kerberos authentication while creating an operational database using COD CLI.

### COD supports creating edge nodes while creating an operational database

COD now allows you to create edge nodes while creating an operational database. You can define the number of edge nodes to be created, and COD automatically creates and configures the nodes in your COD cluster.

You can also add additional nodes into your COD cluster or delete an existing one using the edge node instance ID.

For more information, see *Managing edge nodes*.

### COD supports disabling the Kerberos authentication while creating an operational database using COD CLI

COD now allows you to disable the Kerberos authentication while creating an operational database using COD CLI. You can use the `--disable-kerberos` option while running the `create-database` command to disable the Kerberos authentication.

```
cdp opdb create-database --environment-name ENVIRONMENT_NAME --database-name DATABASE_NAME --disable-kerberos
```

#### Related Information

[Managing edge nodes](#)

## November 10, 2022

Cloudera Operational Database (COD) 1.25 version supports creating and updating an operational database using a custom image.

### COD supports custom images for deploying COD clusters

COD now allows you to create or update a database using a custom image. Custom images can be used for various purposes, such as compliance or security requirements. An image catalog is used to hold one or more custom images. You can inherit pre-installed packages or software from the custom image while creating or updating an operational database.

You can also switch an image catalog of an existing operational database. For more information, see *Managing custom images in COD*.

#### Related Information

[Managing custom images in COD](#)

## October 10, 2022

Cloudera Operational Database (COD) 1.24 version supports fast SSD volume types, deploys strong meta servers for multiple regions, and provides two new CDL CLI commands.

### COD supports fast SSD based volume types for gateway nodes of HEAVY types

Cloudera Manager and monitoring systems require more resources than a regular deployment for big clusters with a large number of nodes. To support this, now COD supports a fast gp2 storage when you choose a gateway node of HEAVY type.

### COD deploys strong meta servers for multiple regions for Multi-AZ

For Multi-AZ deployments, COD assigns the nodes to multiple regions when multiple strong meta servers exist in the deployment.

### COD provides CDP CLI commands to set the HBase configuration values

COD now supports the CDP CLI commands `update-hbase-configuration` and `describe-hbase-configuration` to update and retrieve the HBase configuration values. For more information, see *describe-hbase-configuration* and *update-hbase-configuration*.

### Related Information

[describe-hbase-configuration](#)

[update-hbase-configuration](#)

## August 10, 2022

Cloudera Operational Database (COD) 1.23 version supports custom EBS volumes for HDFS clusters while creating a database and displays UI notifications on the COD UI.

### COD supports custom EBS volumes for HDFS clusters

COD allows you to customize the attached storage worker node for an HDFS cluster using the `--attached-storage-for-workers` option while creating a database. You can define the EBS volumes for HDFS clusters.

You can also view the customized attached storage worker nodes using the `describe-database` and `list-database` commands. For more information, see [create-database](#).

### COD displays notifications on the COD UI for all the major events

COD now displays notifications about all the major events on the COD UI. For example, auto-scaling events, auto-healing events, and alerting notifications.

The UI notifications is a major step in providing enough information on COD UI so that you do not need to navigate to multiple internal systems used by COD to get the required information as well as be aware of the events happening in the system on an immediate basis.

## June 10, 2022

Cloudera Operational Database (COD) 1.22 version supports HBase Store File Tracking, Multiple Availability zones on AWS, and CDP Control Planes for multiple regions.

### COD now supports the Store File Tracking (SFT) as a limited availability feature

Store File Tracking (SFT) defines a separate, pluggable layer to handle storefile life cycle, and includes the FILE based built-in implementation that avoids internal file rename or move operations while managing the storefiles. This is a critical enablement to deploy HBase over S3 object store, which is known for the lack of atomic renames. COD enables this feature by default for databases deployed on AWS with S3, to mitigate the aforementioned S3 limitation that could cause critical issues for HBase. For more information, see [HBase Store File Tracking](#).

### COD supports Multiple Availability Zones (Multi-AZ) on AWS

COD ensures high availability and fault tolerance using Multi-AZ deployments. A Multi-AZ deployment means that compute infrastructure for HBase's master and region servers are distributed across multiple AZs ensuring that when a single availability zone has an outage, only a portion of Region Servers is impacted and clients automatically switch over to the remaining servers in the available AZs.

Multi-AZ for COD is currently supported only on Amazon Web Services (AWS) environments. For more information, see [Multi-AZ deployment on COD](#).

### COD supports CDP Control Planes for multiple regions

COD now supports CDP Control Plane for eu-1 (Germany) and ap-1 (Australia) regions. Certain countries may have regulatory requirements that limit or forbid specific types of data from being sent or stored outside of the country or region where an organization operates. For this reason Cloudera has introduced new regions in which the Control Plane can run. By choosing a region other than us-west-1, you can ensure that sensitive metadata (such as information about CDP users) does not leave its region or country of origin. For more information, see [CDP Control Plane regions](#).

## April 10, 2022

Cloudera Operational Database (COD) 1.21 version supports enabling the replications for META regions, by default.

### **COD now supports replications for META regions enabled by default**

COD now supports replication for the META regions, which means clients can read the META replicas first before connecting to the primary region server. This reduces the load on the META table because the read META load is distributed across multiple replicas.

COD supports configuring three META replicas which can be deployed in different RegionServers. Client needs to use the COD provided configuration to get the META calls that are distributed across these replicas. This is useful when an application has multiple clients accessing spark, yarn, or phoenix-spark.

## March 10, 2022

Cloudera Operational Database (COD) 1.20 version provides UI enhancements to download the Phoenix client jars. Refer to the following section for more details.

### **COD provides enhanced user interface to download the Phoenix client jar**

COD now provides an enhanced user interface to download the Phoenix client jar from the Phoenix Thick and Phoenix Thin client tabs through a single click.

You can now download the Phoenix client jars with a single click directly from the Phoenix Thick client and Phoenix Thin client tabs in the UI.

## February 10, 2022

Cloudera Operational Database (COD) 1.19 version supports Spark transactional tables using Apache OMID and is also bundled with the HBase version 2.4.6. Refer to the detailed section for more features.

### **COD through Phoenix-Spark connector supports Spark transactional tables using Apache OMID**

COD supports Apache OMID transactional framework that allows Big Data applications to execute ACID transactions on top of Phoenix tables.

The transaction support in COD enables you to perform complex distributed transactions and run atomic database operations, meaning your database operations must either be completed or terminated. A transaction ensures adhering to the ACID properties.

### **COD is now bundled with the HBase version 2.4.6**

COD is now bundled and shipped along with the HBase version 2.4.6 when the CDP Runtime version is 7.2.14.

For a smooth and better functionality, COD is now bundled with the HBase version 2.4.6. You need to upgrade the HBase client version for seamless connectivity.

### **COD supports custom table coprocessors**

COD supports custom table coprocessors, which you can implement and extend from HBase coprocessors' interfaces.

You can add table coprocessors so that HBase can run custom code on the server side against the stored data and filter local minimum or maximum value during ingestion without scanning the entire table. You can use built-in table coprocessors from the upstream HBase releases. For more information, see [Working with custom table coprocessors](#).

### **COD supports RAZ integration from the Runtime version 7.2.11.0**

COD supports RAZ integration from the Runtime version 7.2.11.0. You can grant fine-grained access to directories.

The Ranger Authorization Service (RAZ) is a fine grained authorization service for cloud storage. As a regular individual user or as an HBase user, you can limit the authorization levels in the cloud storage to a directory level. For more information, see [COD integration with RAZ](#).

### **Storefile Tracking (SFT) is available as an optional feature delivered through the Cloudera Operational Database (COD) service**

COD now supports the “Storefile Tracking” (SFT) as an optional feature in Runtime 7.2.14.0.

Storefile Tracking (SFT) changes how HBase manages its files to avoid operations which are known to be suboptimal when using object stores. COD enables this feature for COD databases deployed on AWS which use S3 for HBase storage which will address performance issues known around flushes, compactions, and other HBase operations. For more information, see [HBase Storefile Tracking](#).

### **COD allows to disable the Kerberos authentication temporarily for HBase clients**

COD allows to disable the Kerberos authentication temporarily for HBase clients that run on Cloudera legacy products.

If your client applications are running on Cloudera legacy products, they usually do not have Kerberos authentication enabled. When you try to connect to any COD instance, the connection fails because the COD instances have Kerberos enabled, by default. Now, you can disable Kerberos authentication in your COD instances so that HBase or Phoenix clients can connect seamlessly. For more information, see [Disabling Kerberos authentication for HBase clients](#).

## **December 10, 2021**

Cloudera Operational Database (COD) 1.18 version supports ephemeral storage on Azure.

### **COD supports ephemeral storage on Azure**

COD now supports the configuration of instance storage to cache HBase data stored in block storage. This is only available on AMD instance types.

## **November 10, 2021**

Cloudera Operational Database (COD) 1.17 version is now available as a technical preview on GCP (Google Cloud Platform) and with various performance improvements.

### **COD is available as a Technical Preview feature on GCP**

You can now deploy COD on GCP easily similar to what is available for Amazon Web Services (AWS) and Microsoft Azure.

### **COD automatically improves the performance by 80% when you use AWS S3**

COD now delivers a better performance in S3 because the data loading behaviour from S3 into cache is tuned. This improvement minimizes the cost associated with the S3's high latency to read data.

### **COD improves scalability when using block storage on AWS**

COD now uses larger EBS volumes for the underlying master nodes to provide better scalability.

## **October 10, 2021**

Cloudera Operational Database (COD) 1.16 version supports modified auto-scaling criteria and a built-in coprocessor *AGGREGATEIMPLEMENTATION*.

### COD supports modified auto-scaling criteria

COD now supports an improved auto-scaling algorithm that considers the latency of the user operations. COD now prioritises user operations over system operations that results in a reduced cost of infrastructure with a minor increase in replication.

### COD supports a built-in coprocessor *AGGREGATEIMPLEMENTATION*

COD now supports a built-in coprocessor called *AGGREGATEIMPLEMENTATION* that facilitates aggregation function computations (min, max, sum, avg, median, std) at the region level. This yields better performance because you need not get all the data to perform these calculations. COD enables *AGGREGATEIMPLEMENTATION* by default, and you can use the *AGGREGATIONCLIENT* service in HBase to perform RegionServer side aggregation. For example, row count.

## September 10, 2021

Cloudera Operational Database (COD) 1.15 version provides a new option for the `upgrade-database` command and supports DataHub/OpDB deployment across multiple availability zones.

### The `upgrade-database` command supports a new option to upgrade only the operating system

COD provides a new option `--os-upgrade-only` for the `upgrade-database` command, which you can use to upgrade only the operating system to the latest supported version addressing any CVEs that were patched in the operating system. Running the `upgrade-database` command with this option does not have an effect on the CDP runtime version running on the given cluster.

### COD supports DataHub/OpDB deployment across multiple availability zones (Multi-AZ) to ensure high availability

- COD introduces a technical preview version of Multi-AZ deployment capability and is available by special request through your account team.
- When you enable COD, by default it deploys all the databases across three availability zones identified in the SDX by the provided network subnets. COD CLI beta provides a new `--disable-multi-az` to limit deployments to a single availability zone.

### COD can disable ephemeral storage in CLI

COD provides the ability to disable the ephemeral storage using a new command `--disable-ephemeral-storage`.

### COD provides the ability to list all the HBase snapshots using different filters

COD supports listing all the HBase snapshots created within a database of an environment using these filters, time range, snapshot name, and command ID. For example, you can use the time range filter to obtain the snapshots created within a particular time, the command ID filter to know the status of a snapshot operation.

## August 10, 2021

Cloudera Operational Database (COD) 1.14 version offers multiple new features that includes disengaging autoadmin database function, runtime upgrade, copy tables between COD environments and many more.

### Disengage autoadmin database function

COD can disable the autonomous database functions thereby stops managing a database. When COD disables the autonomous functions, the database cannot leverage it; however the functions are available and accessible as a datahub cluster. Once you disable the autonomous database functions, it is irreversible.

### Cloudera Runtime upgrade with downtime

You can upgrade the Cloudera Runtime installed on the cluster running COD to an advanced one with downtime.

### Copy tables between two COD environments

COD provides a CopyTable utility to copy tables from one COD cluster to another.

### COD supports ephemeral storage on AWS

COD now supports ephemeral storage on AWS for HBase buckets.

### COD provides HDFS as a storage option

COD now provides HDFS as a storage option for COD deployment instead of cloud storage.

### COD uses instance storage for HBase bucket cache

COD now deploys clusters with large block cache on ephemeral storage thereby enhancing the performance of the clusters. This is available only for AWS-based clusters with S3 blob storage.

## July 10, 2021

Cloudera Operational Database (COD) 1.13 version offers multiple new features that includes disengaging autoadmin database function, runtime upgrade, copy tables between COD environments and many more.

### COD supports HDFS for storage on Cloud Block Storage in addition to Cloud Object Storage on AWS & Azure

1. COD now supports HDFS for fast and consistent performance.
2. Cloud Object Storage has a high variance in latency, and sometimes the delay in accessing the data is high. The operational cost in managing such an operational database is expensive, as it reads millions of data per second. COD now deploys HDFS on Cloud Block Storage, which ensures that the performance characteristics are similar to on-prem. This feature simplifies the lift-and-shift efforts to the cloud.
3. This capability includes an updated `update-database` command in CDP CLI Beta to enable tuning of auto-scaling based on HDFS utilization.

### COD promotes custom user tags to CDP CLI Beta

When you create a COD database, COD enables you to supply custom tags for launched hardware while using CDP CLI Beta.

### COD delivers utility to manually copy tables across Kerberos realms without cross-realm Kerberos trust

COD introduces a `CldrCopyTable` utility, which is Cloudera's version of the upstream [CopyTable](#) utility. This utility is an extension of the [COD replication plugin](#) that enables you to copy data cross-realm. This utility enables you to close any replication gaps that might arise depending on how you start or stop replication.

### COD streamlines on-prem replication plugins

When you create a COD database, COD automatically sets configuration properties for the Cloudera replication plugin so that on-prem clusters can replicate into COD databases. COD now automatically configures the OpDB replication plugin and reduces the time needed to move the application into production.

## May 10, 2021

Cloudera Operational Database (COD) 1.11 version now supports Phoenix through Hue web UI.

### Cloudera supports Phoenix when using Hue web UI

Cloudera now supports Phoenix access through the Apache Hue SQL assistant. You can now access Phoenix and query the data from the Hue web UI out of the box and leverage Hue's rich SQL support (for example, type ahead) to query Phoenix for operational insights or developer testing of SQL statements.

### February 10, 2021

Cloudera Operational Database (COD) now supports deploying a data hub cluster as an edge node.

### COD now supports deploying a data hub cluster as an edge node

You can deploy a Data Hub cluster that works as an edge node to access your COD instance. Use the cluster template [\*\*\**RUNTIME VERSION*\*\*\*] COD Edge Node cluster template. For example, 7.2.7 COD Edge Node. For more information see [Configure Data Hub edge node](#).

### November 10, 2020

Cloudera Operational Database (COD) is now available as a technical preview.

### COD is available as a technical preview

COD is now available as a technical preview and is considered under development. For an overview about this service, see [Cloudera Operational Database service in the public cloud](#).



**Important:** Do not use this in your production systems. If you have feedback, contact Support by logging a case on the Cloudera Support Portal at <https://my.cloudera.com/support.html>. Technical preview services and features are not guaranteed troubleshooting and fixes.

## Fixed issues

Learn about the fixed issues and the enhancements in Cloudera Operational Database (COD).

### COD-3836 (COD 1.41) : COD supports deployments to specific subnets in Azure and GCP environments

When you create an operational database using the CDP CLI, the `subnet-id` parameter in the `create-database` command is fixed and can be used to deploy the cluster into specific subnets in Azure and GCP environments.

Following is an example of `subnet-id`.

```
cdp opdb create-database --environment-name <env> --database-name <db> --subnet-id DataLakeHubSubnet1
```

For more information, see *CDP CLI documentation*.

### ODX-2928 (COD 1.31) : Minor upgrade is not available

Minor upgrades are available now while using the `describe-updgrade-database` CDP CLI command. You can now list the available hotfix versions using the `describe-updgrade-database` CDP CLI command.

### ODX-1716 (COD 1.21) : Increase storage density for HDFS form factor

Storage density for HDFS form factor is increased. Now, COD deploys 8 TB of disks for each worker instead of 6 TB during any new HDFS deployment.

### ODX-1811 (COD 1.21) : Increase handler count

The number of RegionServer handlers is increased to process 50% more read or write requests.



## Related Information

[CDP CLI documentation](#)

# Known issues

Learn about the known issues in the Cloudera Operational Database, the impact or changes to the functionality, and the workaround.

## **COD-3987 (COD 1.46): LB based client configs and tarball URLs in the describe-client-connectivity command output fails intermittently**

**Problem:** Due to a known Cloudbreak issue, CB-26030, downloading an HBase client configuration intermittently fails when using multiple Gateway nodes

**Workaround:** Retry the download operation until it succeeds.

## **COD-3611 (COD 1.38): [Azure] Import TSV job during AZ outage fails to write data to the HBase table**

**Problem:** COD is unable to write data on an HBase table and also fails to import a TSV file during a Multi-AZ outage on an Azure environment.

**Workaround:** None

## **ODX-1603 (COD 1.22) : COD throws a "Node Failure" issue**

**Problem:** COD reports "Node Failure" state whenever OMID service goes down and retains the state until the OMID service is restarted from Cloudera Manager (CM).

**Workaround:**

- If you are using OMID, restart the OMID service through CM and wait for a few minutes until COD starts displaying Available again.
- If you are not using OMID service and your COD status is not affected by the OMID service status, you can suppress the OMID alerts directly in the CM.

## Technical Service Bulletins

### **TSB 2022-568: HBase normalizer must be disabled for Salted Phoenix tables**

When Apache Phoenix (“Phoenix”) creates a salted table, it pre-splits the table according to the number of salt regions. These regions must always be kept separate, otherwise Phoenix does not work correctly.

The HBase normalizer is not aware of this requirement, and in some cases the pre-split regions are merged automatically. This causes failure in Phoenix.

The same requirement applies when merging regions of salted tables manually: regions containing different salt keys (the first byte of the rowkey) must never be merged.

Note that either automatic or manual splitting of the regions for a salted table does not cause a problem. The problem only occurs when adjacent regions containing different salt keys are merged.

## Upstream JIRA

[PHOENIX-4906](#)

## Knowledge article

For the latest update on this issue, see the corresponding Knowledge article: [TSB 2022-568: Hbase normalizer must be disabled for Salted Phoenix tables](#)