

Custom Table Coprocessors in Cloudera Operational Database (Preview)

Date published: 2022-02-10

Date modified: 2023-08-10

Legal Notice

© Cludera Inc. 2025. All rights reserved.

The documentation is and contains Cludera 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 Cludera software may be found within the documentation accompanying each component in a particular release.

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

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

Cludera, Cludera Altus, HUE, Impala, Cludera Impala, and other Cludera 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.

This document has been released as part of a technical preview for features described herein. Technical preview components are provided as a convenience to our customers for their evaluation and trial usage. These components are provided 'as is' without warranty or support. Further, Cludera assumes no liability for the usage of technical preview components, which should be used by customers at their own risk.

Contents

| | |
|---|----------|
| Legal Notice | 2 |
| Contents | 3 |
| Custom table coprocessors in Cloudera Operational Database | 4 |
| Adding custom coprocessors into HBase tables | 4 |
| Removing custom coprocessors from HBase tables | 5 |
| Obtaining a list of coprocessors from a COD environment | 5 |
| Verifying table coprocessors in HBase tables | 6 |

This document has been released as part of a technical preview for features described herein. Technical preview components are provided as a convenience to our customers for their evaluation and trial usage. These components are provided 'as is' without warranty or support. Further, Cloudera assumes no liability for the usage of technical preview components, which should be used by customers at their own risk.

Custom table coprocessors in Cloudera Operational Database

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 values during ingestion without scanning the entire table.

You can use built-in table coprocessors from the upstream HBase releases. COD supports custom table coprocessors, which you can implement and extend from HBase coprocessors' interfaces.

You can add custom table coprocessors into the HBase tables or remove them from HBase tables, also obtain a list of coprocessors with its status on the COD environment.

You must ensure that:

- You have the CDP CLI setup.
- You have uploaded the JAR file (custom coprocessor) to a remote location that the COD instance can access.

Important:

This feature is in Technical Preview and is not ready for production deployment. Cloudera encourages you to explore these technical preview features in non-production environments and provide feedback on your experiences.

Related information

- [Coprocessor Introduction](#)

Adding custom coprocessors into HBase tables

Learn how to add custom table coprocessors to the HBase table.

Steps

1. Launch the CDP CLI tool.
2. Upload your custom coprocessor to a remote location. For example, S3.

This document has been released as part of a technical preview for features described herein. Technical preview components are provided as a convenience to our customers for their evaluation and trial usage. These components are provided 'as is' without warranty or support. Further, Cloudera assumes no liability for the usage of technical preview components, which should be used by customers at their own risk.

3. Run the following **add-coprocessor** command to add a specific coprocessor using its canonical class name.

```
# add table-level coprocessor
cdp opdb add-coprocessor \
--environment ENVIRONMENT \
--database DATABASE \
--table-name TABLE_NAME \
--coprocessor-canonical-name COPROCESSOR_CANONICAL_NAME \
[--coprocessor-location-url COPROCESSOR_LOCATION_URL] \
[--coprocessor-args COPROCESSOR_ARGS]
```

Removing custom coprocessors from HBase tables

Learn how to remove the custom coprocessors from the HBase tables.

Steps

1. Launch the CDP CLI tool.
2. Run the following **remove-coprocessor** command to remove a specific coprocessor using its canonical class name.

```
# remove table-level coprocessor
cdp opdb remove-coprocessor \
--environment ENVIRONMENT \
--database DATABASE \
--table-name TABLE_NAME \
--coprocessor-canonical-name COPROCESSOR_CANONICAL_NAME \
[--force] \
[--no-force]
```

Obtaining a list of coprocessors from a COD environment

Learn how to obtain a list of coprocessors from a COD environment.

This document has been released as part of a technical preview for features described herein. Technical preview components are provided as a convenience to our customers for their evaluation and trial usage. These components are provided 'as is' without warranty or support. Further, Cloudera assumes no liability for the usage of technical preview components, which should be used by customers at their own risk.

Steps

1. Launch the CDP CLI tool.
2. Run the following **list-coprocessors** command to obtain a list of all the added coprocessors.

```
# Get a list of coprocessors with its status on the COD
environment
cdp opdb list-coprocessors \
--environment ENVIRONMENT \
--database DATABASE \
[--table-name TABLE_NAME] \
[--command-id COMMAND_ID]
```

Verifying table coprocessors in HBase tables

After you add or remove the table coprocessors through CDP CLI, you can verify whether they are added successfully.

Steps

1. Open the HBase UI from the Cloudera Manager.
2. Click **User Tables** under **Tables**.
3. Check the coprocessors' details in the **Description** column for the respective HBase tables.