

Cloudera Runtime 7.3.1

Atlas Ozone

Date published: 2020-07-28

Date modified: 2024-12-10

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

About Apache Ozone integration with Apache Atlas.....	4
How Integration works.....	4

About Apache Ozone integration with Apache Atlas

When you integrate Ozone with Apache Atlas, entities like Apache Hive, Spark processes, and NiFi flows, when created with Ozone path, results in Atlas creating Ozone path entities.

Integration supports the Ozone location of Ozone-backed Hive set-up. The event which triggers an Ozone entity to be created in Atlas is a DDL query in Hive which is backed by Apache Ozone.

To learn more about Apache Ozone, see [Apache Ozone documentation](#).

Currently, integrating Atlas with Ozone enables creation of specific Ozone entities in Atlas. Apache Ozone is an object store for Hadoop Data Lake Workloads, similar to HDFS, Microsoft ABFS, and Amazon S3.

Ozone provides three main abstractions:

- **Volumes:** Volumes are similar to a home directory. Volumes are used to store buckets. Only administrators can create or delete volumes. Once a volume is created, users can create as many buckets as needed.
- **Buckets:** Buckets are similar to directories. A bucket can contain any number of keys, but buckets cannot contain other buckets. Ozone stores data as keys which live inside these buckets.
- **Keys:** Keys are similar to files.

Previously, in Atlas for Cloudera, Hive entities created with Ozone path resulted in the creation of HDFS path entities.

When a Hive external entity was created with a Ozone path, for example: ofs://ozone1/volume1/bucket1/file. It results in file creation at the HDFS path in Atlas.

**Note:**

Before Cloudera on premises version 7.1.9 SP1 CHF3, OFS paths could not be parsed by Ranger TagSync, hence tag based policies cannot be used with Atlas for tables with OFS keys.

If you are already using existing Ozone keys (either OFS or O3FS) which are already tagged and have tag based policies defined before upgrading to 7.1.9 SP1 CHF3 or later versions (except version 7.3.1), retag these keys in Atlas after upgrading (Delete the tags and reapply them).

For more information, see [What's new in Cloudera Runtime 7.1.9 SP1 CHF3](#).

How Integration works

This integration mechanism does not provide a direct Atlas Hook or Atlas Bridge option for Ozone to listen to the entity events in Ozone.

Because Atlas does not have the direct hook and Hive only provides path information, Atlas populates only a few attributes for Ozone entities.

Consider an example use case to create an external table with an Ozone path that results in creating an Ozone path entity directly in Atlas. In the example, the name of the Hive database is “oz_demo”.

Using the cluster interface, run the following command:

```
$ CREATE EXTERNAL TABLE sales (id int, name string) row format delimited fields terminated by ' ' stored as textfile location 'ofs://ozone1/volume1/bucket1/table_oz_demo1';
```



Note:
Before Cloudera on premises version 7.1.9 SP1 CHF3, OFS paths could not be parsed by Ranger TagSync, hence tag based policies cannot be used with Atlas for tables with OFS keys.

If you are already using existing Ozone keys (either OFS or O3FS) which are already tagged and have tag based policies defined before upgrading to 7.1.9 SP1 CHF3 or later versions (except version 7.3.1), retag these keys in Atlas after upgrading (Delete the tags and reapply them).

For more information, see [What's new in Cloudera Runtime 7.1.9 SP1 CHF3](#).

Once the query is executed, a table named “table_oz_demo1” is created in Atlas Web UI.

oz_demo (hive_db)

Classifications:

Terms:

PropertiesRelationshipsClassificationsAudits**Tables**

☐ Exclude sub-types

☐ Exclude sub-classifications

☐ Show historical entities

☐	Name	Owner	Description	Type	Classifications	Term
☐	table_oz_demo_1	hive		hive_table		

Open the External table (table_oz_demo1) and select the Lineage tab to display the path from where the table was created in Atlas.

table_oz_demo_1 (hive_table)

Classifications:

Terms:

Properties**Lineage**RelationshipsClassificationsAuditsSchema

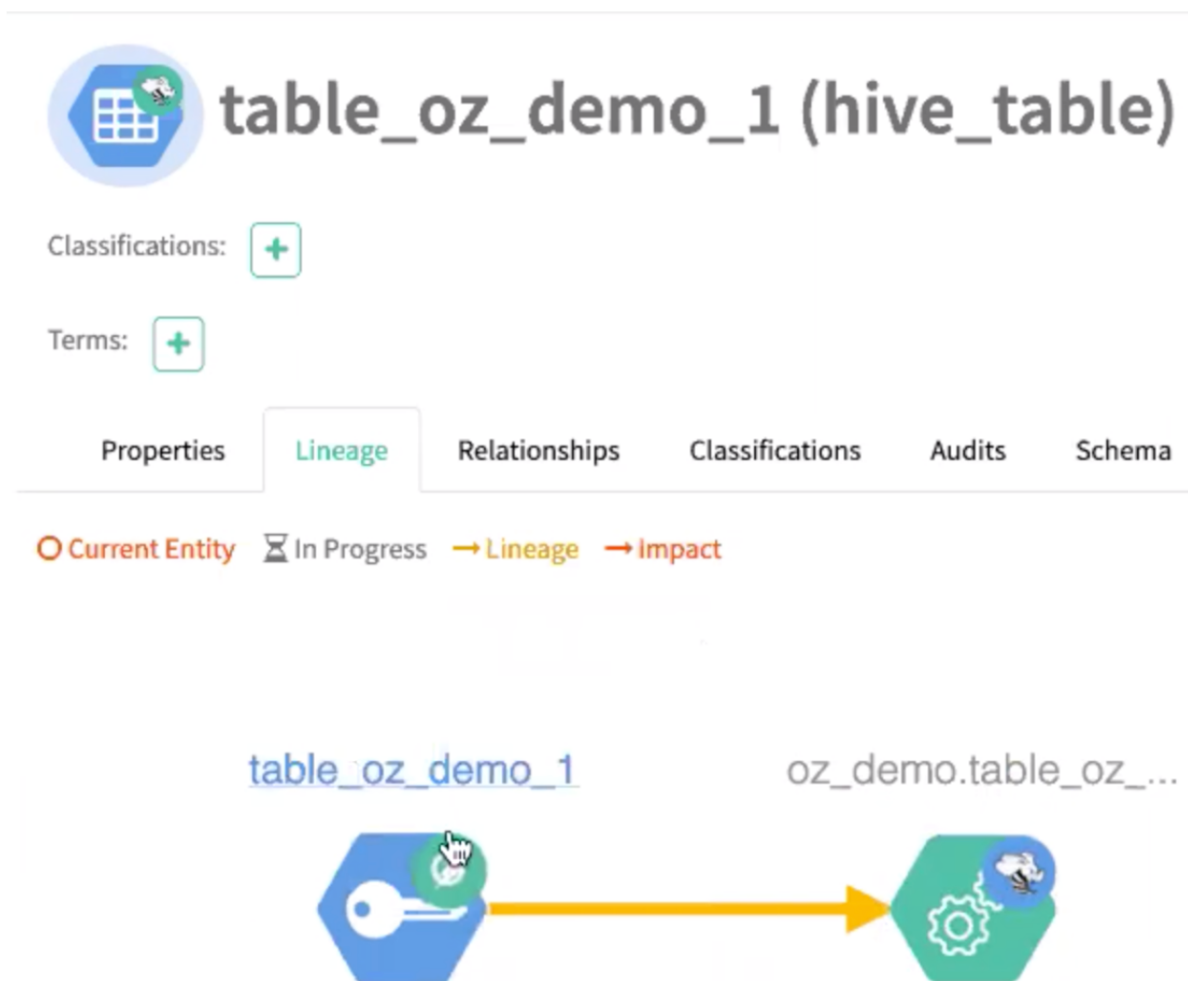
Current EntityIn ProgressLineageImpact

table_oz_demo_1

oz_demo.table_oz_...

table_oz_demo_1

Select the Ozone key (table_oz_demo_1) to view the information in the slider window.



The slider displays the typeName as the Ozone key.

table_oz_demo_1 (hive_table)

Classifications:

Terms:

Properties | Lineage | Relationships | Classifications | Audits | Schema

Current Entity In Progress → Lineage → Impact

```

graph LR
    A[oz_demo.table_oz_demo_1] --> B[Hive Engine Icon]
    B --> C[table_oz_demo_1]
  
```

Key	Value
guid	de8dfa82-de53-4fec-8ca4-681189d31ac2
typeName	ozone_key
name	table_oz_demo_1
qualifiedName	o3fs://bucket1.volume1.ozone1/table_oz_demo_1@cm
status	ACTIVE
classifications	N/A

The Ozone Key refers to the Ozone path entity. The Ozone Key here is under “bucket1”.

How Ozone integration is set up

The following image provides an insight into the Atlas-Ozone integration mechanism.

```
{
  "name": "ozone_volume",
  "superTypes": [
    "DataSet"
  ],
  ...
}

{
  "name": "ozone_bucket",
  "superTypes": [
    "DataSet",
    "ozone_parent"
  ],
  ...
}

{
  "name": "ozone_key",
  "superTypes": [
    "DataSet",
    "ozone_parent",
    "ozone_child"
  ],
  ...
}
```

