Cloudera Runtime 7.1.6

Atlas Ozone

Date published: 2021-02-29
Date modified: 2021-03-19

https://docs.cloudera.com/

Legal Notice

Contents

About Apache Ozone integration with Apache Atlas

When you integrate Ozone with Atlas, entities like Hive, Spark process, 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 know more about Apache Ozone, see Apache Hadoop Ozone.

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 CDP, 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: o3fs://bucket1.volume1.ozone1/file. It resulted in the creation of the HDFS path in Atlas.

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.

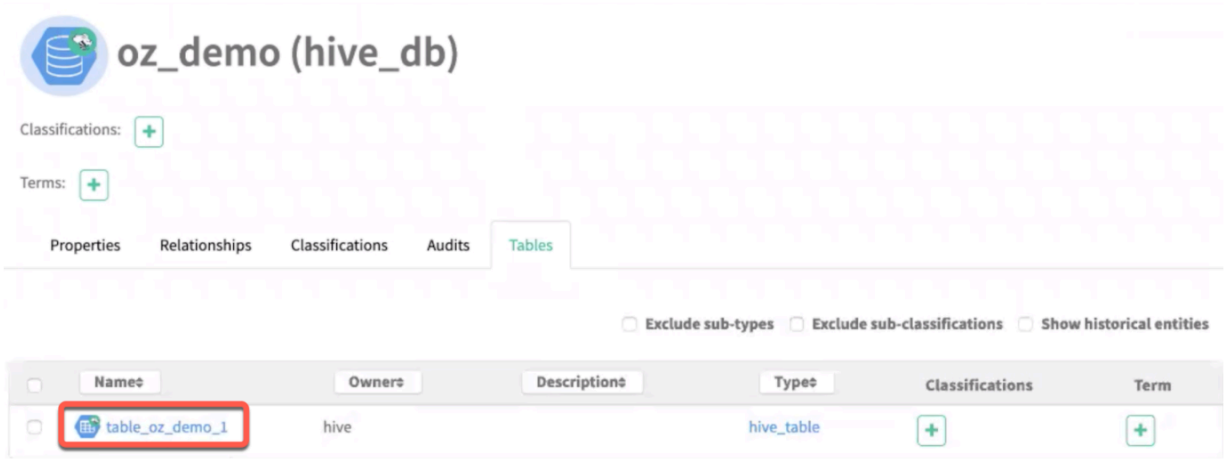
Because Atlas does not have the direct hook and Hive only provides path information, Atlas populates only a few attributes for Ozone entities.

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 'o3fs://bucket1.volume1.ozone1/table_oz_demo1';

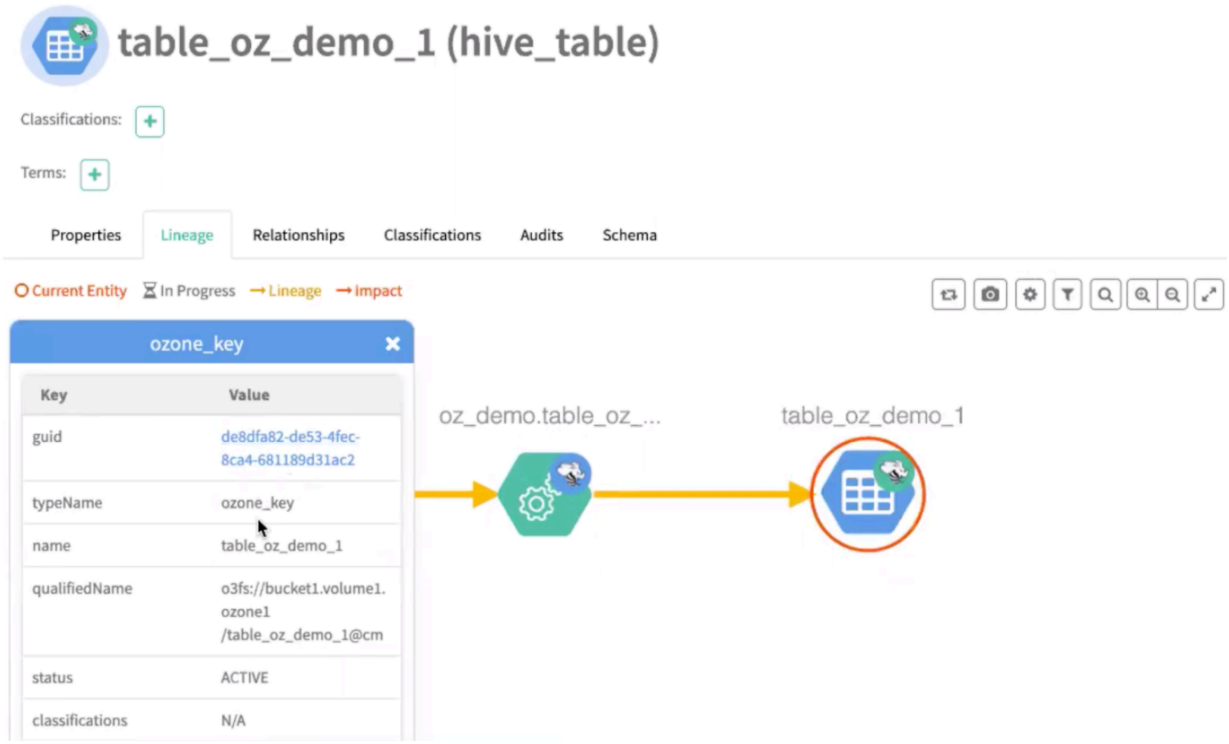
Once the query is executed, a table named “table_oz_demo1” is created in Atlas Web UI.



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

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.



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"
    ],
    ...
}
```