Cloudera Data Flow for Data Hub 7.2.1

Analyzing data with Apache Hbase in CDP Public Cloud

Date published: 2020-06-23
Date modified: 2020-06-23

CLOUDERA

https://docs.cloudera.com/

Legal Notice

Contents

Analyzing your data with HBase

After preparing your environment, you need to choose a source to which you connect Flink in Data Hub. After generating data to your source, Flink applies the computations you have added in your application design. The results are redirected to your HBase sink.

Before you begin

- You have a CDP Public Cloud environment.
- You have a CDP username (it can be your own CDP user or a CDP machine user) and a password set to access Data Hub clusters.

  The predefined resource role of this user is at least EnvironmentUser. This resource role provides the ability to view Data Hub clusters and set the FreeIPA password for the environment.
- Your user is synchronized to the CDP Public Cloud environment.
- You have a Streaming Analytics cluster.
- You have an Operational Database with SQL cluster in the same Data Hub environment as the Streaming Analytics cluster.
- Your CDP user has the correct permissions set up in Ranger allowing access to HBase.

Procedure

1. Choose a source for your Flink application and add the connector to your application.

   Note: You can create your application choosing the Streams Messaging cluster with Kafka, Schema Registry and Streams Messaging Manager. For more information about Kafka as a source, see the Analyzing your data with Kafka use case.
2. Add HBase as sink to your Flink application.

   The following code example shows how to build your application logic with an HBase sink:

```
HBaseSinkFunction<QueryResult> hbaseSink = new HBaseSinkFunction<QueryR
esult>("ITEM_QUERIES") {
@Override
public void executeMutations(QueryResult qresult, Context context, Buffer
edMutator mutator) throws Exception {
  Put put = new Put(Bytes.toBytes(qresult.queryId));
  put.addColumn(Bytes.toBytes("itemId"), Bytes.toBytes("str"),   Bytes.to
Bytes(qresult.itemInfo.itemId));
  put.addColumn(Bytes.toBytes("quantity"), Bytes.toBytes("int"), Bytes.
toBytes(qresult.itemInfo.quantity));
  mutator.mutate(put);
  }
};
hbaseSink.setWriteOptions(HBaseWriteOptions.builder()
  .setBufferFlushIntervalMillis(1000)
  .build()
);
streamqueryResultStream.addSink(hbaseSink);
```

3. Start generating data to your source connector.
4. Deploy your Flink streaming application.

What to do next

You have the following options to monitor and manage your Flink applications:

Job monitoring with Flink Dashboard

After submitting a Flink job, you can always use the Flink Dashboard to review if the job submission was successful. Later you can use the Flink Dashboard to monitor the history of all your submitted and completed jobs.

You can access the Flink Dashboard directly from your Data Hub cluster.

1. Go to Management Console > Data Hub Clusters.
2. Search for your Streaming Analytics cluster.
3. Select Flink Dashboard from the list of Services.

You are redirected to the Flink Dashboard user interface where you can select from the submitted jobs.

Note: You cannot save the Completed jobs into cloud storage.

Metadata governance with Atlas

You can use Atlas to find, organize and manage different assets of data about your Flink applications and how they relate to each other. This enables a range of data stewardship and regulatory compliance use cases.

Procedure

1. Go to Management Console > Data Lakes .
2. Search for your environment from the list of available environments.
3. Select Atlas from the list of services.
4. Search and select flink_application from the Search By Type bar.
5. Click the Name of your application.
6. Select Properties, Lineage, Relationships, Classifications or Audits tabs for more information about your application.