Quickstart

This guide provides a quick peek at Hudi's capabilities using spark-shell. Using Spark datasources, we will walk through code snippets that allows you to insert and update a Hudi dataset of default storage type: Copy on Write. After each write operation we will also show how to read the data both snapshot and incrementally.

Setup spark-shell

Hudi works with Spark-2.x versions. You can follow instructions here for setting up spark.

From the extracted directory run spark-shell with Hudi as:

```
bin/spark-shell --packages org.apache.hudi:hudi-spark-bundle:0.5.0-incubating --conf 'spark.s
erializer=org.apache.spark.serializer.KryoSerializer'
```

Setup table name, base path and a data generator to generate records for this guide.

```
import org.apache.hudi.QuickstartUtils._
import scala.collection.JavaConversions._
import org.apache.spark.sql.SaveMode._
import org.apache.hudi.DataSourceReadOptions._
import org.apache.hudi.DataSourceWriteOptions._
import org.apache.hudi.config.HoodieWriteConfig._

val tableName = "hudi_cow_table"
val basePath = "file:///tmp/hudi_cow_table"
val dataGen = new DataGenerator
```

The DataGenerator can generate sample inserts and updates based on the the sample trip schema here

Insert data

Generate some new trips, load them into a DataFrame and write the DataFrame into the Hudi dataset as below.

Docker Demo

A Demo using docker containers

Lets use a real world example to see how hudi works end to end. For this purpose, a self contained data infrastructure is brought up in a local docker cluster within your computer.

The steps have been tested on a Mac laptop

Prerequisites

- Docker Setup : For Mac, Please follow the steps as defined in [https://docs.docker.com/v17.12/docker-for-mac/install/]. For running Spark-SQL queries, please ensure atleast 6 GB and 4 CPUs are allocated to Docker (See Docker -> Preferences -> Advanced). Otherwise, spark-SQL queries could be killed because of memory issues.
- kafkacat : A command-line utility to publish/consume from kafka topics. Use `brew install kafkacat` to install kafkacat
- /etc/hosts : The demo references many services running in container by the hostname. Add the following settings to /etc/hosts

```
127.0.0.1 adhoc-1
127.0.0.1 adhoc-2
127.0.0.1 namenode
127.0.0.1 datanode1
127.0.0.1 hiveserver
127.0.0.1 hivemetastore
127.0.0.1 kafkabroker
127.0.0.1 sparkmaster
127.0.0.1 zookeeper
```

Also, this has not been tested on some environments like Docker on Windows.

Setting up Docker Cluster

Build Hudi

The first step is to build hudi

Docker Demo

A Demo using docker containers

Lets use a real world example to see how hudi works end to end. For this purpose, a self contained data infrastructure is brought up in a local docker cluster within your computer.

The steps have been tested on a Mac laptop

Prerequisites

- Docker Setup : For Mac, Please follow the steps as defined in [https://docs.docker.com/v17.12/docker-for-mac/install/]. For running Spark-SQL queries, please ensure atleast 6 GB and 4 CPUs are allocated to Docker (See Docker -> Preferences -> Advanced). Otherwise, spark-SQL queries could be killed because of memory issues.
- kafkacat : A command-line utility to publish/consume from kafka topics. Use `brew install kafkacat` to install kafkacat
- /etc/hosts : The demo references many services running in container by the hostname. Add the following settings to /etc/hosts

```
127.0.0.1 adhoc-1
127.0.0.1 adhoc-2
127.0.0.1 namenode
127.0.0.1 datanode1
127.0.0.1 hiveserver
127.0.0.1 hivemetastore
127.0.0.1 kafkabroker
127.0.0.1 sparkmaster
127.0.0.1 zookeeper
```

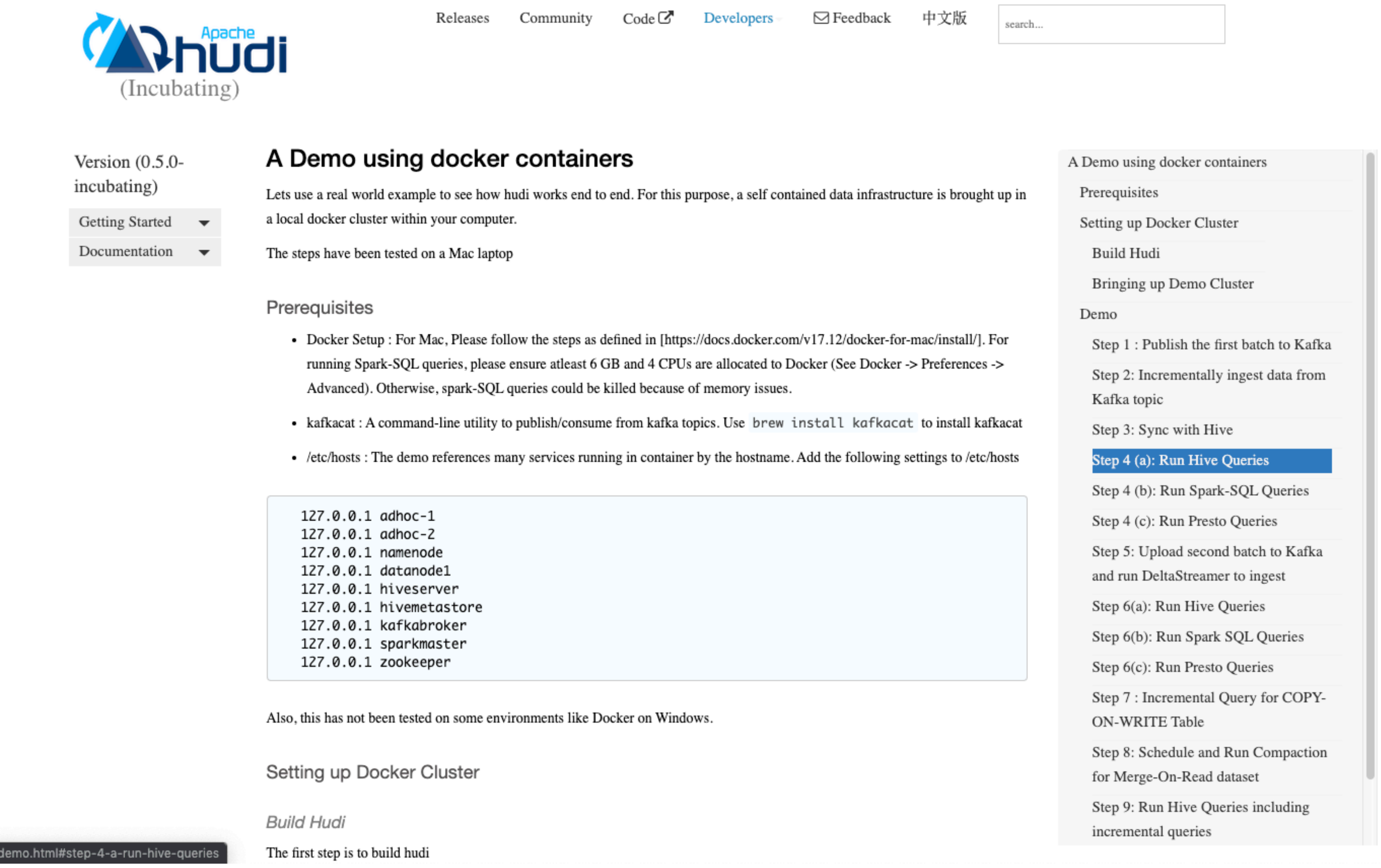
Also, this has not been tested on some environments like Docker on Windows.

Setting up Docker Cluster

Build Hudi

The first step is to build hudi

tep-5-upload-second-batch-to-kafka-and-run-deltastreamer-to-ingest



A Demo using docker containers

Lets use a real world example to see how hudi works end to end. For this purpose, a self contained data infrastructure is brought up in a local docker cluster within your computer.

The steps have been tested on a Mac laptop

Prerequisites

- Docker Setup : For Mac, Please follow the steps as defined in [https://docs.docker.com/v17.12/docker-for-mac/install/]. For running Spark-SQL queries, please ensure atleast 6 GB and 4 CPUs are allocated to Docker (See Docker -> Preferences -> Advanced). Otherwise, spark-SQL queries could be killed because of memory issues.
- kafkacat : A command-line utility to publish/consume from kafka topics. Use `brew install kafkacat` to install kafkacat
- /etc/hosts : The demo references many services running in container by the hostname. Add the following settings to /etc/hosts

```
127.0.0.1 adhoc-1
127.0.0.1 adhoc-2
127.0.0.1 namenode
127.0.0.1 datanode1
127.0.0.1 hiveserver
127.0.0.1 hivemetastore
127.0.0.1 kafkabroker
127.0.0.1 sparkmaster
127.0.0.1 zookeeper
```

Also, this has not been tested on some environments like Docker on Windows.

Setting up Docker Cluster

Build Hudi

The first step is to build hudi

demo.html#step-4-a-run-hive-queries