**Cloudera Search Full Text Search with Solr**

**HBase NRT Indexer and Search**

**Author: Michelle Zhang**

**Company: Cloudera**

**Date: 10/29/2014**

# Install Cloudera Search Related Components in Cluster

## Install Solr Related Components

After you installed CM, you can use CM “Add a Service” entry to automatically install Solr Related Components.

Base Components: ZooKeeper, HDFS, YARN(MR2 included), HBase

Cloudera Search Need Components: Solr, Flume, Key-Value Store Indexer, Hue

## Start Solr Related Components’ Services

Start Solr related components’ services in sequence with CM:

CM 🡪 Home 🡪 Start Services(Solr, Flume, Key-Value Store Indexer, Hue)

Take CDH 5.2 cluster as an example, and set $HOME=/root/solr, we will introduce how to deploy a NRT(Near Real Time) indexer and search in CDH Cloudera Search.

# Enable Cluster-wide HBase Replication

## Configure Replication in CM

According to /opt/cloudera/parcels/CDH/share/doc/hbase-solr-doc-1.5+cdh5.2.0+19/demo/hbase-site.xml, set the properties values in CM🡪 HBase 🡪 Confiuration. 🡪 “Service-Wide”

You must add these settings to all of the hbase-site.xml configuration files on the HBase cluster, except the replication.replicationsource.implementation property.

## Restart HBase Services in CM

Restart HBase services by CM.

# Point Lily HBase NRT Indexer Service

## Add Properties in hbase-indexer-site.xml

Add following properties in /etc/hbase-solr/conf/hbase-indexer-site.xml.

# cat /etc/hbase-solr/conf/hbase-indexer-site.xml

<property>

<name>hbase.zookeeper.quorum</name>

<value>ip-172-31-12-149.us-west-2.compute.internal,ip-172-31-12-150.us-west-2.compute.internal,ip-172-31-12-151.us-west-2.compute.internal</value>

</property>

<property>

<name>hbaseindexer.zookeeper.connectstring</name>

<value>ip-172-31-12-149.us-west-2.compute.internal,ip-172-31-12-150.us-west-2.compute.internal,ip-172-31-12-151.us-west-2.compute.internal:2181</value>

</property>

## Scp to Other Two Nodes

# scp /etc/hbase-solr/conf/hbase-indexer-site.xml root@172.31.12.150:/etc/hbase-solr/conf/

# scp /etc/hbase-solr/conf/hbase-indexer-site.xml [root@172.31.12.151:/etc/hbase-solr/conf/](mailto:root@172.31.12.151:/etc/hbase-solr/conf/)

## Restart Lily HBase services

Restart Lily HBase service by CM.

## Check Lily HBase NRT Indexer Services

# jps –lm

34106 sun.tools.jps.Jps -lm

32382 com.ngdata.hbaseindexer.Main

# Enable Replication for HBase Column Families

For every existing table, set the REPLICATION\_SCOPE on every column family that needs to be indexed.

# disable 'record'

#alter 'record', {NAME => 'data', REPLICATION\_SCOPE => 1}

# enable 'record'

# Create Collection for HBase NRT Indexer and Search

## The section is same as “HBase Batch Indexer and Search” doc.

## Generate Configuration for Collection

Generate the configuration files for the collection:

# solrctl instancedir --generate /root/solr/hbase-collection1

## Prepare Schema.xml for collection

Edit schema.xml in the conf dir to configure Fields, UniqueKey and CopyField:

<fields>

<field name="data" type="text\_general" indexed="true" stored="true" required="true" multiValued="false" />

<field name="id" type="string" indexed="true" stored="true" required="true" />

<field name="text" type="text\_general" indexed="true" stored="true" multiValued="true"/>

<field name="\_version\_" type="long" indexed="true" stored="true"/>

</fields>

<uniqueKey>id</uniqueKey>

<copyField source="data" dest="text"/>

<copyField source="id" dest="text"/>

## Upload Instance Directory to ZooKeeper

# solrctl instancedir --create hbase-collection1 /root/solr/hbase-collection1

## Create New Collection in HDFS

# solrctl collection --create hbase-collection1

Then you can verify whether the collection is live in Solr Server Web UI.

For example: <http://ec2-54-69-194-12.us-west-2.compute.amazonaws.com:8983/solr/#/~cloud>



## Create Lily HBase Indexer Configuration

# cat /root/solr/morphline-hbase-mapper.xml

<?xml version="1.0"?>

<indexer table="record" mapper="com.ngdata.hbaseindexer.morphline.MorphlineResultToSolrMapper">

<!-- The relative or absolute path on the local file system to the morphline configuration file. -->

<!-- Use relative path "morphlines.conf" for morphlines managed by Cloudera Manager -->

<param name="morphlineFile" value**="/etc/hbase-solr/conf/morphlines.conf**"/>

<!-- The optional morphlineId identifies a morphline if there are multiple morphlines in morphlines.conf -->

<!-- <param name="morphlineId" value="morphline1"/> -->

</indexer>

Notice: Keep the value **"/etc/hbase-solr/conf/morphlines.conf**” for “morphlineFile”.

## Create Morphline

It is related to Lily HBase -> Configurations->Service-Wide->Morphlines. But no need to configure it when do batch indexing.

# cat /etc/hbase-solr/conf/morphlines.conf

SOLR\_LOCATOR : {

# Name of solr collection

collection : hbase-collection1

# ZooKeeper ensemble

zkHost : "$ZK\_HOST"

}

morphlines : [

{

id : morphline1

importCommands : ["org.kitesdk.\*\*", "com.ngdata.\*\*"]

commands : [

{

**extractHBaseCells** {

mappings : [

{

inputColumn : "data:\*"

outputField : "data"

type : string

source : value

}

]

}

}

{ logDebug { format : "output record: {}", args : ["@{}"] } }

]

}

]

The morphlines.conf configuration file must be present on every host that runs an indexer.

## Restart Lily HBase services

Restart Lily HBase service by CM.

# Add Lily HBase NRT Indexer

## Register Lily HBase Indexer Configuration

hbase-indexer add-indexer --name myIndexer \

--indexer-conf /root/solr/morphline-hbase-mapper.xml \

--connection-param solr.zk=$ZKHOST/solr \

--connection-param solr.collection=hbase-collection1 \

--zookeeper $ZKHOST:2181

## Verify Lily HBase NRT Indexer

[root@ip-172-31-12-149 conf]# hbase-indexer list-indexers

ZooKeeper connection string not specified, using default: localhost:2181

Number of indexes: 1

myIndexer

+ Lifecycle state: **ACTIVE**

+ Incremental indexing state: SUBSCRIBE\_AND\_CONSUME

+ Batch indexing state: INACTIVE

+ SEP subscription ID: Indexer\_myIndexer

+ SEP subscription timestamp: 2014-09-30T08:03:22.961Z

+ Connection type: solr

+ Connection params:

+ solr.collection = **hbase-collection1**

+ solr.zk = **ip-172-31-12-149.us-west-2.compute.internal,ip-172-31-12-150.us-west-2.compute.internal,ip-172-31-12-151.us-west-2.compute.internal/solr**

+ Indexer config:

576 bytes, use -dump to see content

+ Indexer component factory: com.ngdata.hbaseindexer.conf.DefaultIndexerComponentFactory

+ Additional batch index CLI arguments:

(none)

+ Default additional batch index CLI arguments:

(none)

+ Processes

**+ 1 running processes**

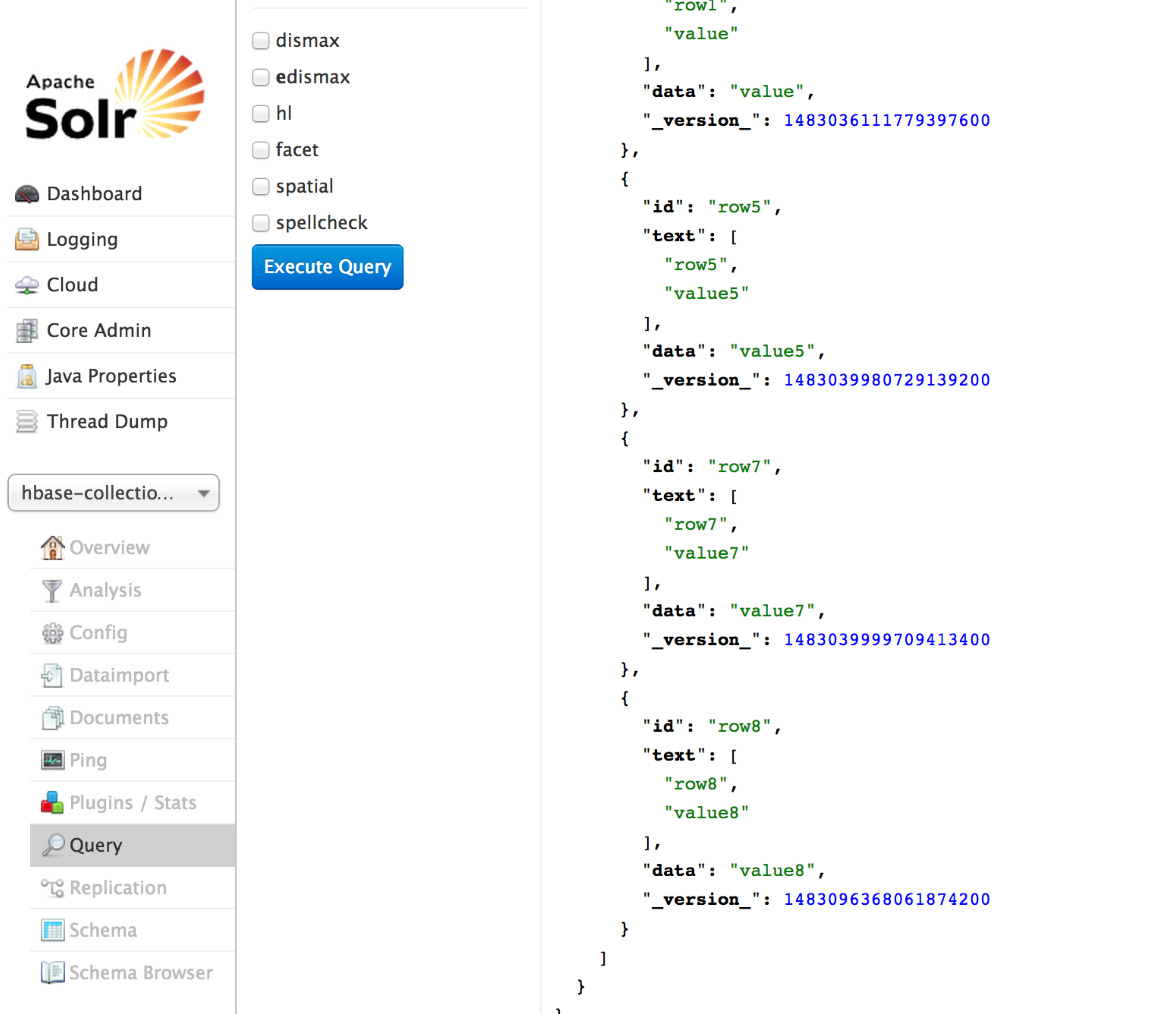
+ 0 failed processes

# Cloudera Search Result in Solr and Hue UI

## Check Solr Queries

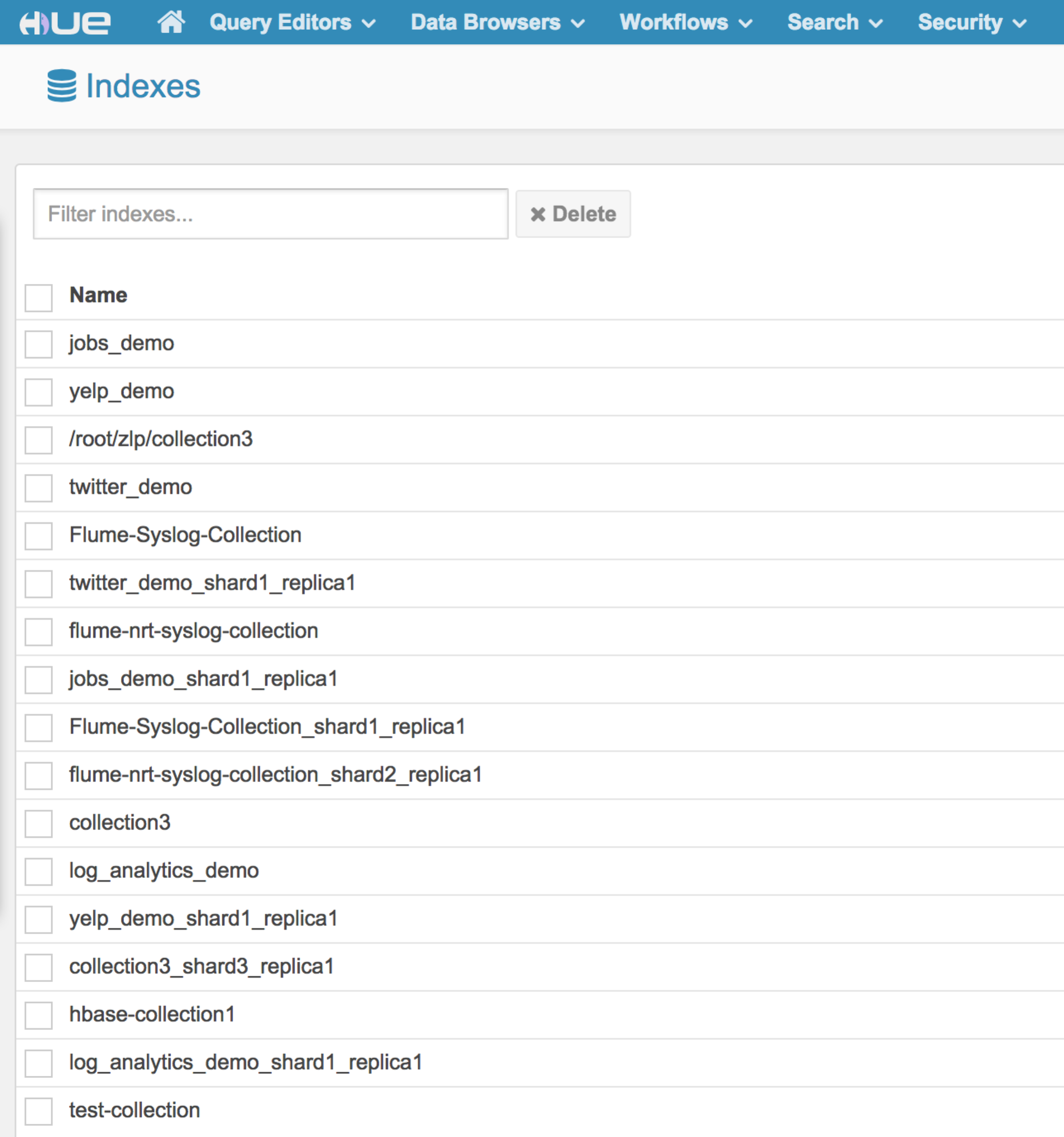
You can check Solr Server Web UI to query the results

<http://ec2-54-69-242-17.us-west-2.compute.amazonaws.com:8983/solr/#/hbase-collection1_shard1_replica1/query>



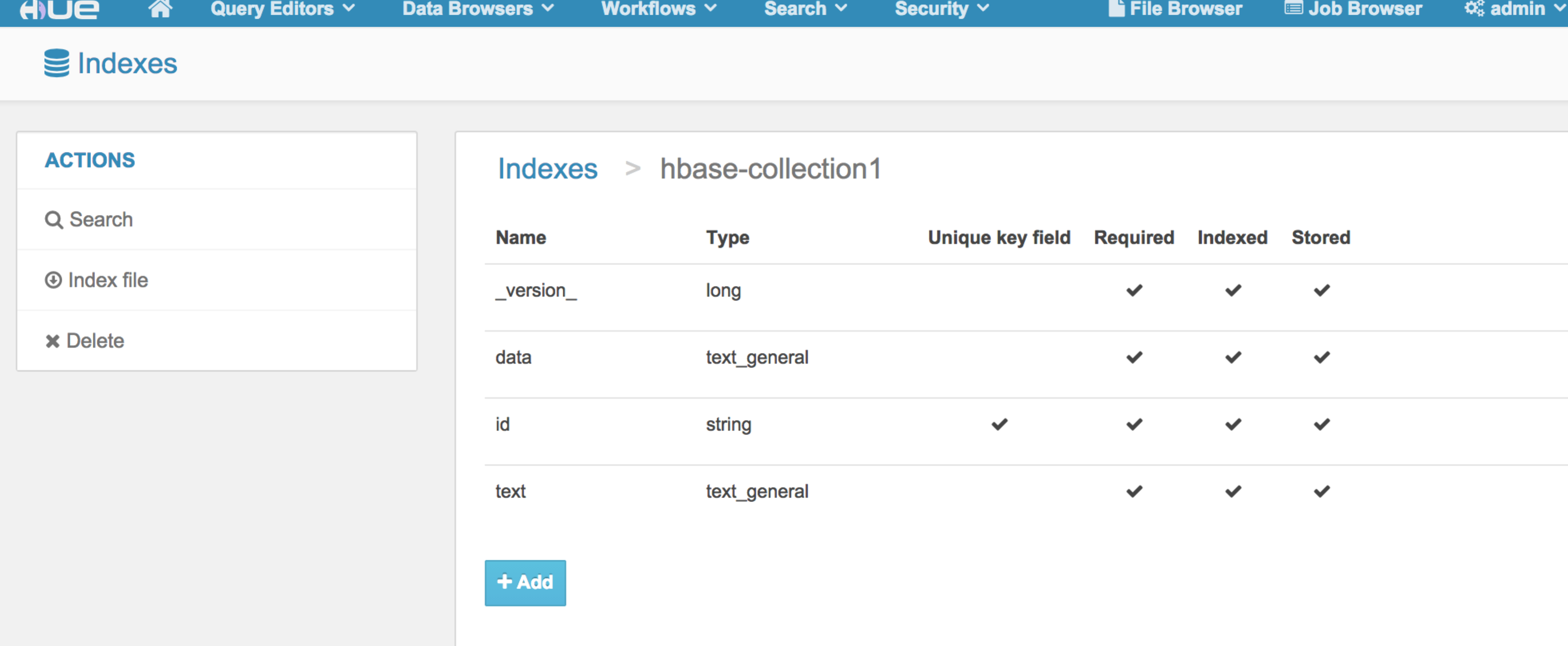
## Hue → Search 🡪 Indexes

Hue 🡪 Search 🡪 Indexes, you can see all the available indexes in Cloudera Search.



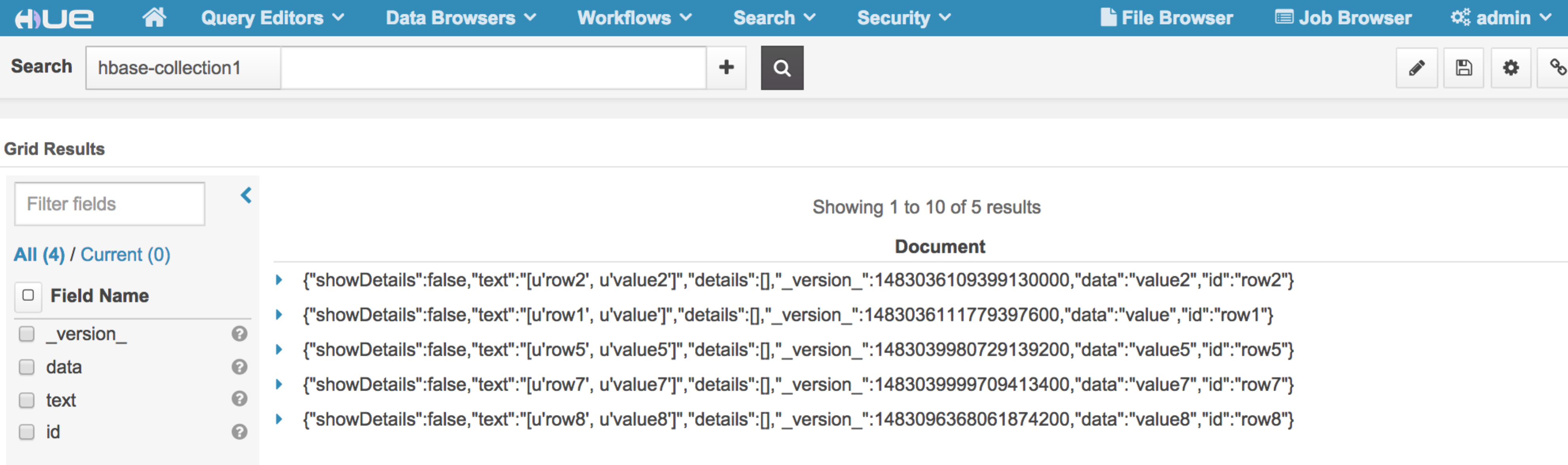
## Hue → Search 🡪 Indexes 🡪 hbase-collection1

Hue 🡪 Search 🡪 Indexes🡪hbase-collection1, you can see all the available indexes definition for hbase-collection1.



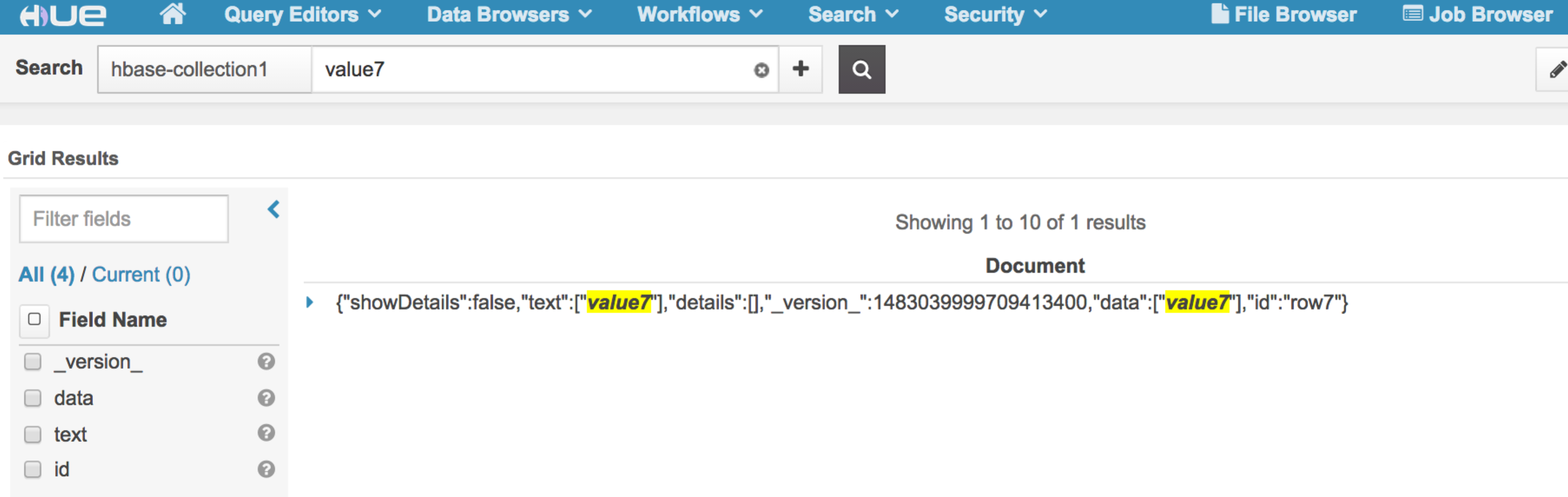
## Hue → Search 🡪 Indexes 🡪 hbase-collection1🡪 Search

Hue 🡪 Search 🡪 Indexes🡪 hbase-collection1🡪 Search, you can see the Search Bar and, index Fields, and results for hbase-collection1.



## Search Keyword

Hue 🡪 Search 🡪 Indexes🡪 hbase-collection1🡪 Search, Search keyword in Search Bar to run full text search, for example: “value7”.



## Verify NRT Indexer and Search

To verify the index is updated in NRT, add new data into table to verify NRT indexing

Add rows to the indexed HBase table:

# put 'record', 'row9', 'data', 'value9'

# put 'record', 'row10', 'data', 'value10'

## Search Keyword

Hue 🡪 Search 🡪 Indexes🡪 hbase-collection1🡪 Search, Search keyword in Search Bar to run full text search, for example: “row9”.

You should be able to immediately search for that message.

