Continuing the Journey on DataFrames & Spark SQL

Xavier Morera

HELPING DEVELOPERS UNDERSTAND SEARCH & BIG DATA

@xmorera www.xaviermorera.com

Do you know what a DSL is?

Just checking up

Domain Specific Language

- Language designed for a specific purpose
- Dataframes are schema aware
- Expose a rich domain specific language
- Structured data manipulation
- SQL like way
- "Think" in SQL

```
Select * From Posts
```

```
postsDF.show()

postsDF.select("*").show()
```

```
Select id From Posts
```

```
postsDF.select("Id").show()
```

```
Select id From Posts
where id = 1
```

```
postsDF.select("id")
.where(col("id") === 1).show()
```

```
postsDF.select("Id")
postsDF.select("Id").show()
postsDF.select("Id").show(5)
postsDF.select("Id").limit(5).show()
postsDF.select("Id", "Title").show(5)
```

Querying DataFrames

So far we have been using show() and select()

- Control how many rows displayed, can use also limit()
- Which columns as well

```
postsDF.select("Id").show(1)

postsDF.select("Id", "Title").show(1)

postsDF.select("Id", "Title", $"Score").show(1)

postsDF.select(col("Id"), postsDF("Title"), $"Score",
  $"CreationDate").show(1)

postsDF.select(col("Id"), postsDF("Title"), $"Score" * 1000,
  $"CreationDate").show(1)
```

Querying DataFrames

Multiple columns

- Several different notations available, careful with not mixing them

Perform an operation, remember: column expression

```
postsDF.filter(col("PostTypeId") === 1).count()
postsDF.where(col("PostTypeId") === 1).count()
postsDF.select(col("PostTypeId")).distinct().show()
```

Filtering Data

Use filter() to specify a condition

Or where(), which is just an alias for filter()

Other functions like distinct()

```
postsDF.select($"Id", when(col("PostTypeId") === 1,
  "Question").otherwise("Other"), $"Title").show(5)

postsDF.select($"Id", when(col("PostTypeId") === 1,
  "Question").otherwise("Other").alias("PostType"), $"Title").show(5)

postsDF.select($"Id", when(col("PostTypeId") === 1,
  "Question").when(col("PostTypeId") === 2,
  "Answer").otherwise("Other").alias("PostType"), $"Title").show(100)
```

Evaluating Conditions

Use when() and otherwise()

Remember that conditions need to be columns

Use alias() for column names

```
postsDF.where((col("PostTypeId") === 5) || (col("PostTypeId") === 1)).count()
postsDF.where((col("PostTypeId") === 5) && (col("PostTypeId") === 1)).count()
```

Multiple Conditions in Expression

Use | and & to specify a Boolean condition

```
val qDF = questionsDF.select("Title", "AnswerCount", "Score")
qDF.orderBy("AnswerCount").show(5)
qDF.orderBy($"AnswerCount".desc).show(5)
qDF.sort($"AnswerCount".desc).show(5)
```

Ordering Results

Use orderBy() or its alias, sort()

Use asc() or desc()

Now you know what a DSL is

Or at least the most important parts

Handling Null Values

DataFrameNAFunctions

- Remove rows with null values
- Fill specific value

Multiple parameters

- Drop if all/any values are null
- Specific columns

```
postsDF.select("Id", "Title").na.drop("any").show(10)

postsDF.select("Id", "Title").na.drop("all").show(10)

postsDF.select("Id", "Title").na.drop("any",
  Seq("Title")).show(10)
```

Removing Nulls

Use na.drop()

Parameters

- how: any or all
- cols: rows to drop based on nulls as specified in how

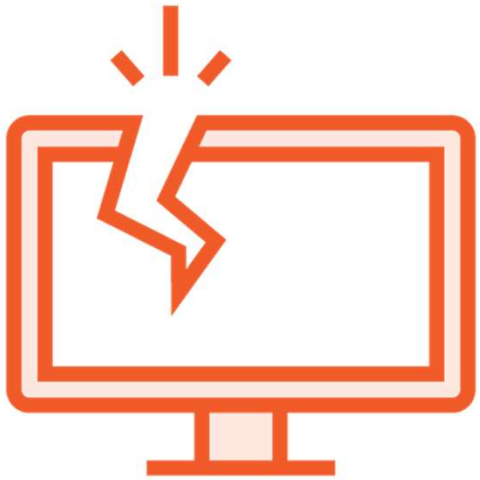
```
postsDF.select("Id", "Title").na.replace(Seq("Title"),
  Map("How can I add line numbers to Vim?" ->
  "[Redacted]")).show(5)

postsDF.select("Id", "Title").na.fill("[N/A]").show(5)
```

Replacing & Filling

Replace a value

Fill null values

Corrupt Records



Invalid data is a possibility

Corrupt records

Multiple ways of handling this scenario

- Separate "bad data"
- Drop record altogether
- Raise exception

```
val badges_for_JSON = badges_columns_rdd.
  map(x => (x(0), x(1), x(2), x(3), x(4), x(5)))

val badgeColumns =
  Seq("Id", "UserId", "Name", "Date", "BadgeClass", "TagBase")

badges_for_JSON.toDF(badgeColumns: _*).write.
  json("/user/cloudera/stackexchange/badges_records")

// Here I corrupt manually the first record
```

```
val badgesDF =
spark.read.json("/user/cloudera/stackexchange/badges_records")

badgesDF.show(5)
```

Create a Bad Record

Take badges

Save as JSON

"Corrupt" the first record

Query | Search data and saved documents... | Jobs | hdfs

File Browser

View as binary

Download

View file location

Refresh

Last modified
01/31/2018
11:12 PM

User
hdfs

Group
supergroup

Size
999.9 KB

Mode
100644

Home

Page 1 to 50 of 250

/ user / cloudera / stackexchange / badges_records /
part-00000-721f757d-8a45-4db1-be54-3a1644b32a5f-c000.json


```
{"_1":"1","_2":"5","_3":"Autobio this is a corrupt record,"_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"2","_2":"4","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"3","_2":"3","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"4","_2":"2","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"5","_2":"1","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"6","_2":"2","_3":"Student","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"7","_2":"5","_3":"Supporter","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"8","_2":"2","_3":"Precognitive","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"9","_2":"5","_3":"Precognitive","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"10","_2":"7","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"11","_2":"6","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"12","_2":"2","_3":"Supporter","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"13","_2":"5","_3":"Teacher","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"14","_2":"6","_3":"Informed","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"15","_2":"17","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
{"_1":"16","_2":"15","_3":"Autobiographer","_4":"2015-02-03T00:00:00.000Z","_5":"3","_6":"False"}
```

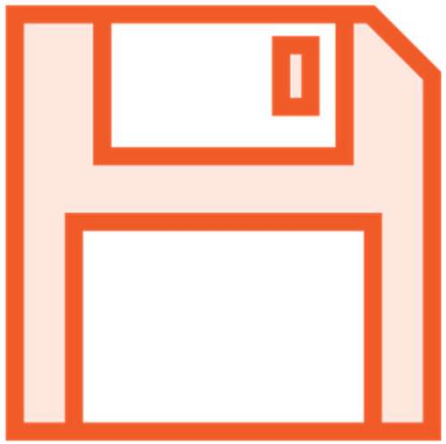
```
spark.read.option("mode",
"PERMISSIVE").json("/user/cloudera/stackexchange/badges_records").show(5)

spark.read.option("mode", "PERMISSIVE").option("columnNameOfCorruptRecord",
"Invalid").json("/user/cloudera/stackexchange/badges_records").show(5)

spark.read.option("mode",
"DROPMALFORMED").json("/user/cloudera/stackexchange/badges_records").show(5)

spark.read.option("mode",
"FAILFAST").json("/user/cloudera/stackexchange/badges_records").show(5)
```

Handling of Corrupt Records

Row with null values

Specify mode: PERMISSIVE, DROPMALFORMED, or FAILFAST

Saving DataFrames



Created DataFrames in a few different ways

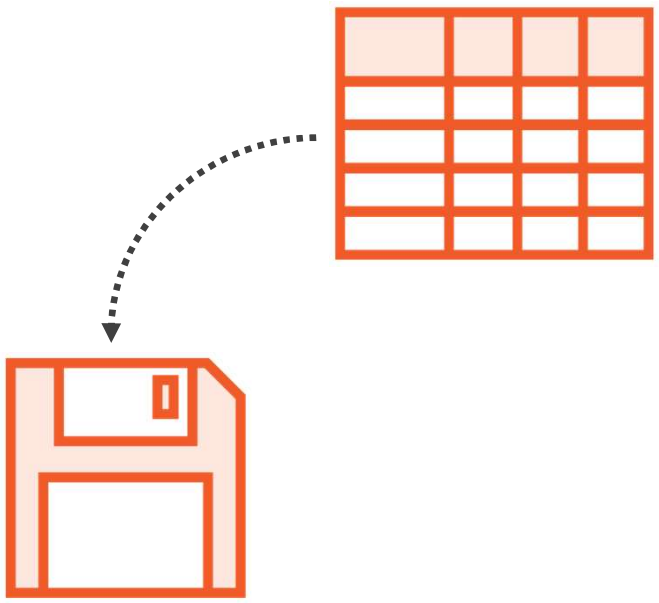
- Returned data back to the Driver
- Persisted into storage
- Supported file format
- Storage

Earlier we learned how to save RDDs

It is a pretty "similar" process With a few differences

Saving DataFrames

DataFrameWriter



Save a DataFrame

- Result of a query
- With write() method

Default is parquet

- Just like RDDs, also configurable

Multiple formats

More than just disk

- Save as a table

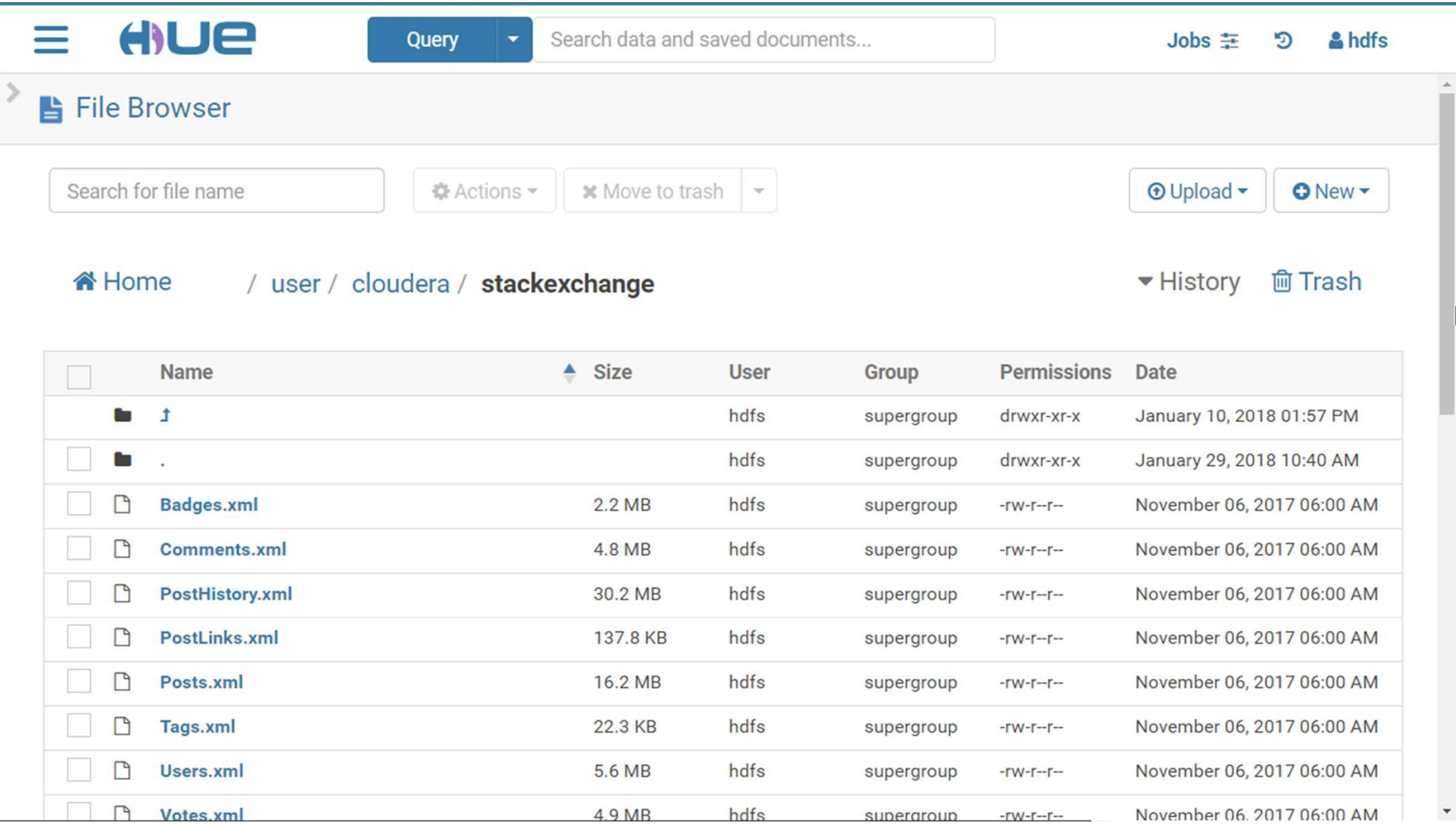
```
postsDF.write.save("/user/cloudera/stackexchange/dataframes/
just_save")

postsDF.rdd.getNumPartitions
```

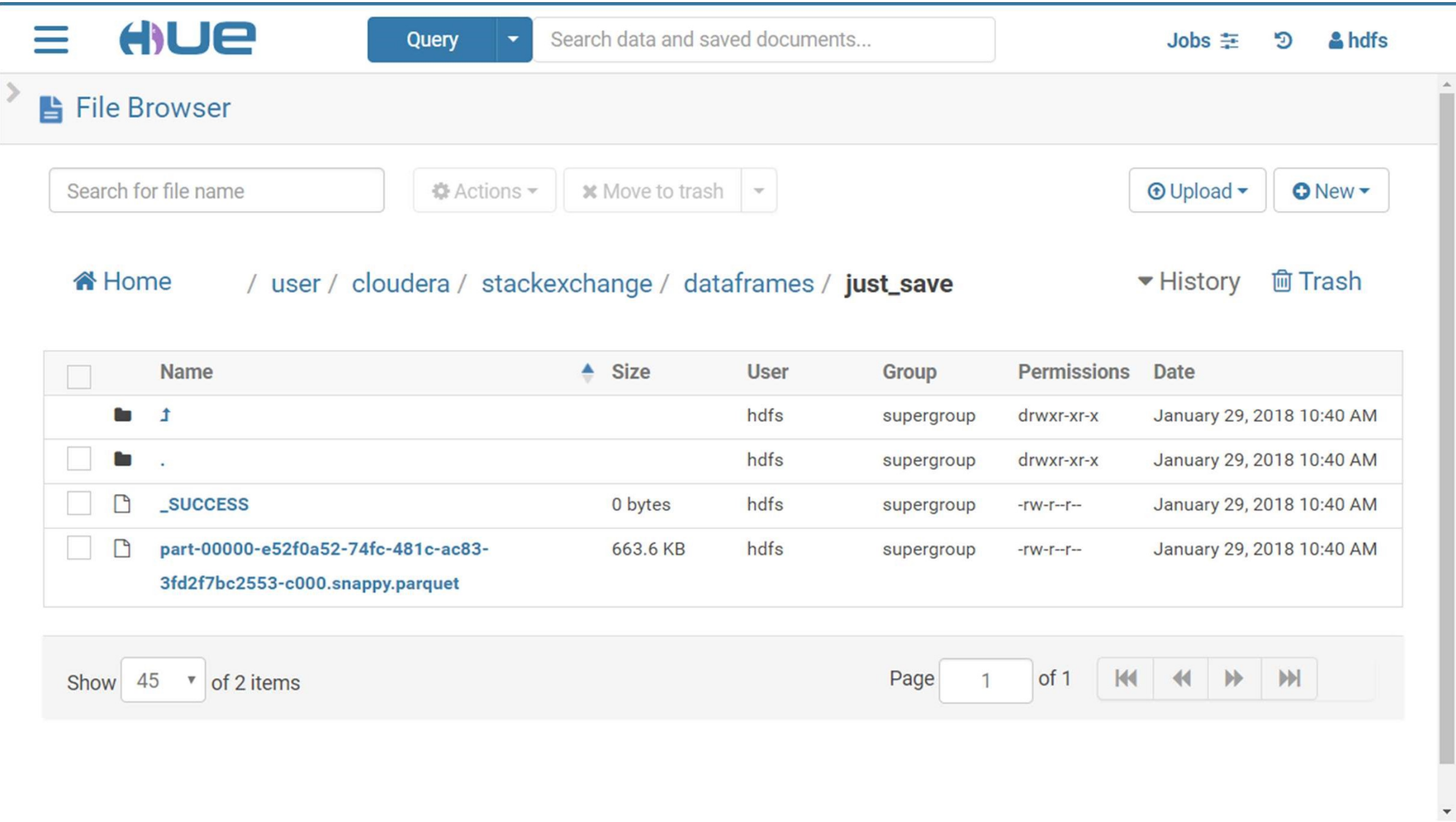
Save – No Options

Use write.save()

Parquet as default

What about partitions?


Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 10, 2018 01:57 PM
. hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
Badges.xml 2.2 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
Comments.xml 4.8 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
PostHistory.xml 30.2 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
PostLinks.xml 137.8 KB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
Posts.xml 16.2 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
Tags.xml 22.3 KB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
Users.xml 5.6 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM
Votes.xml 4.9 MB hdfs supergroup -rw-r--r-- November 06, 2017 06:00 AM

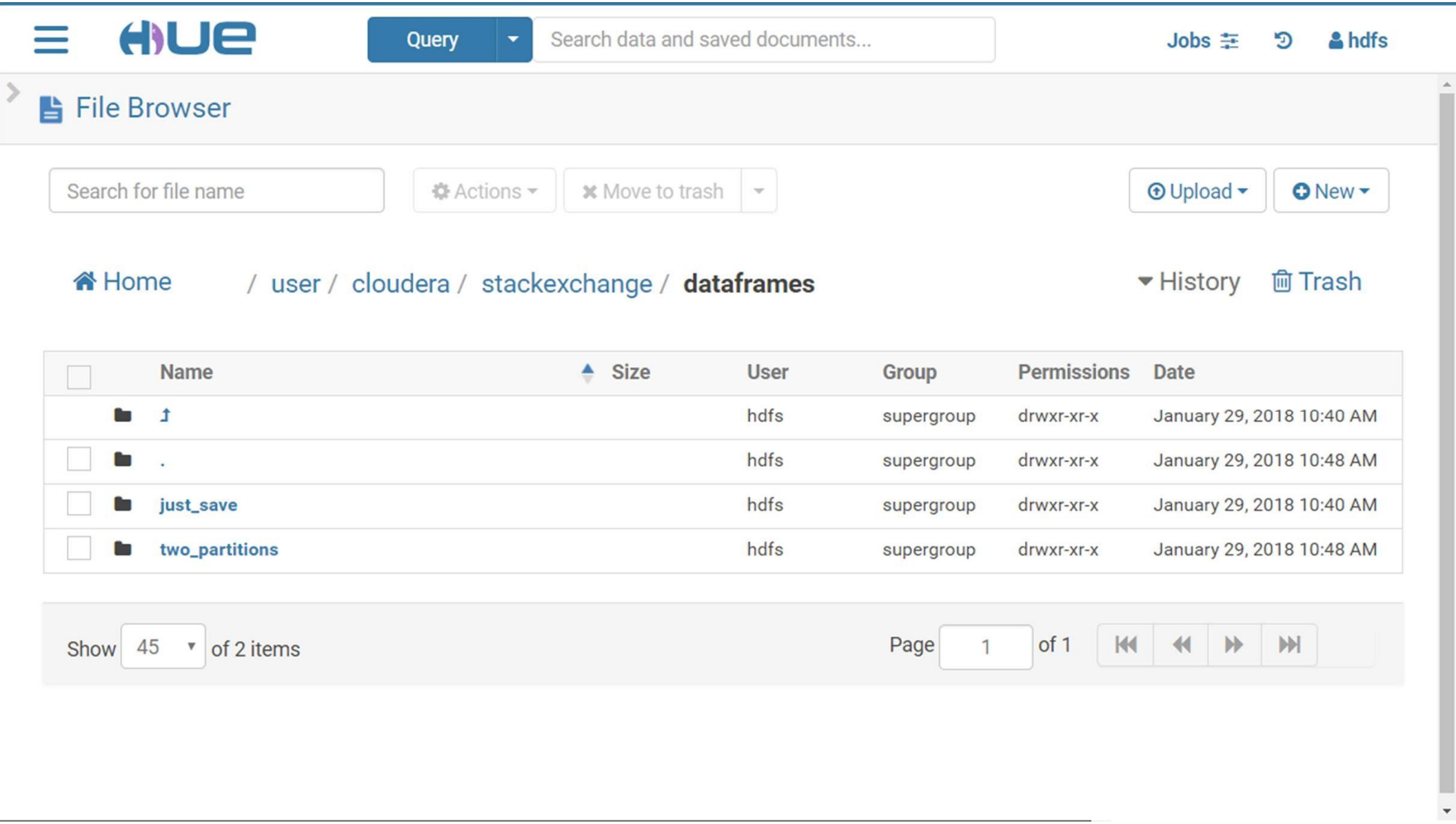

HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange / dataframes / just_save
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
. hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
_SUCCESS 0 bytes hdfs supergroup -rw-r--r-- January 29, 2018 10:40 AM
part-00000-e52f0a52-74fc-481c-ac83-3fd2f7bc2553-c000.snappy.parquet 663.6 KB hdfs supergroup -rw-r--r-- January 29, 2018 10:40 AM
Show 45 of 2 items
Page 1 of 1

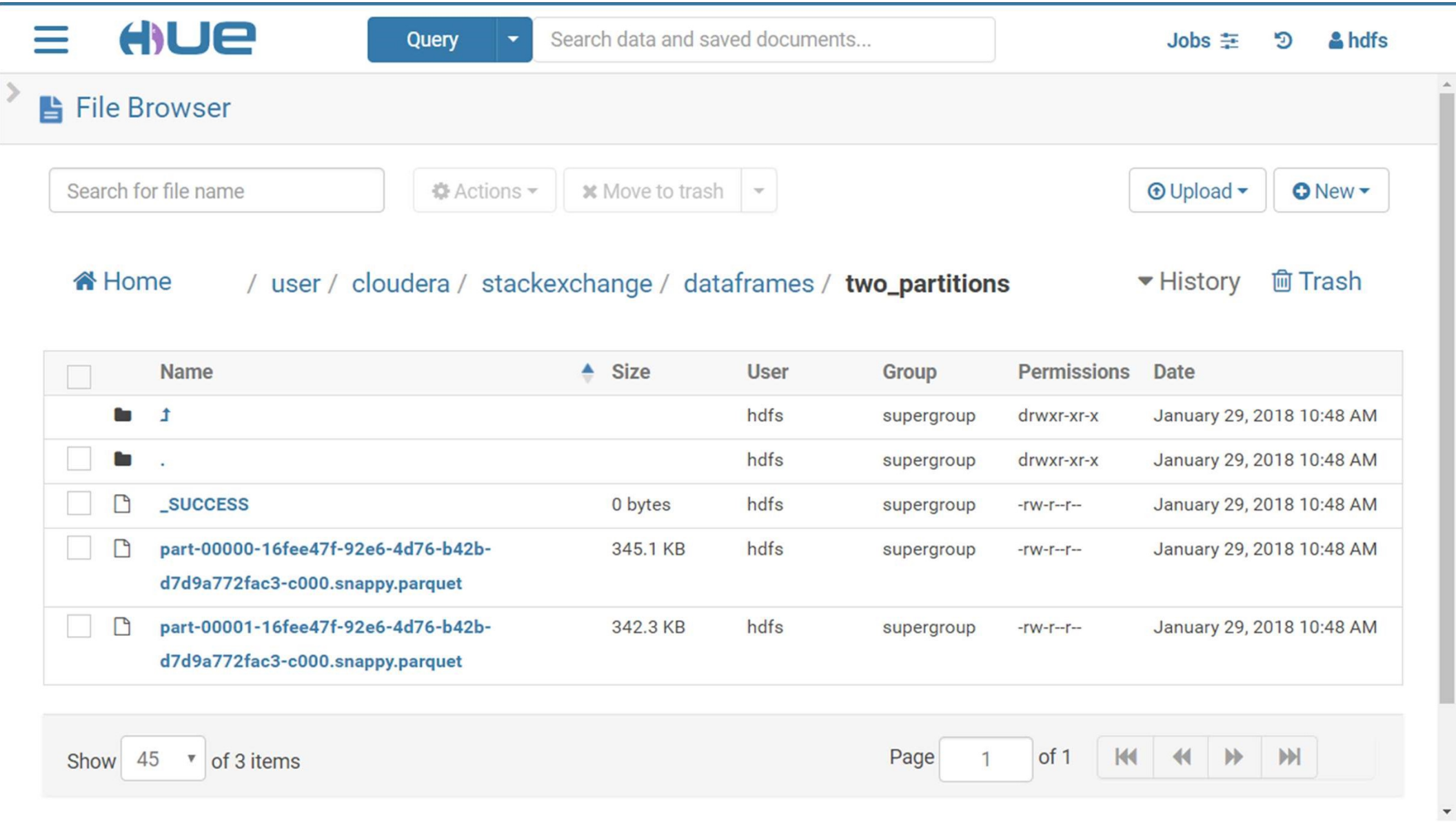
```
postsDF.repartition(2).write.save("/user/cloudera/
  stackexchange/dataframes/two_partitions")
```

Save – No Options

Repartition

Check the results


HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange / dataframes
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
. hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
just_save hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
two_partitions hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
Show 45 of 2 items
Page 1 of 1


Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange / dataframes / two_partitions
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
_SUCCESS 0 bytes hdfs supergroup -rw-r--r-- January 29, 2018 10:48 AM
part-00000-16fee47f-92e6-4d76-b42b-d7d9a772fac3-c000.snappy.parquet 345.1 KB hdfs supergroup -rw-r--r-- January 29, 2018 10:48 AM
part-00001-16fee47f-92e6-4d76-b42b-d7d9a772fac3-c000.snappy.parquet 342.3 KB hdfs supergroup -rw-r--r-- January 29, 2018 10:48 AM
Show 45 of 3 items
Page 1 of 1

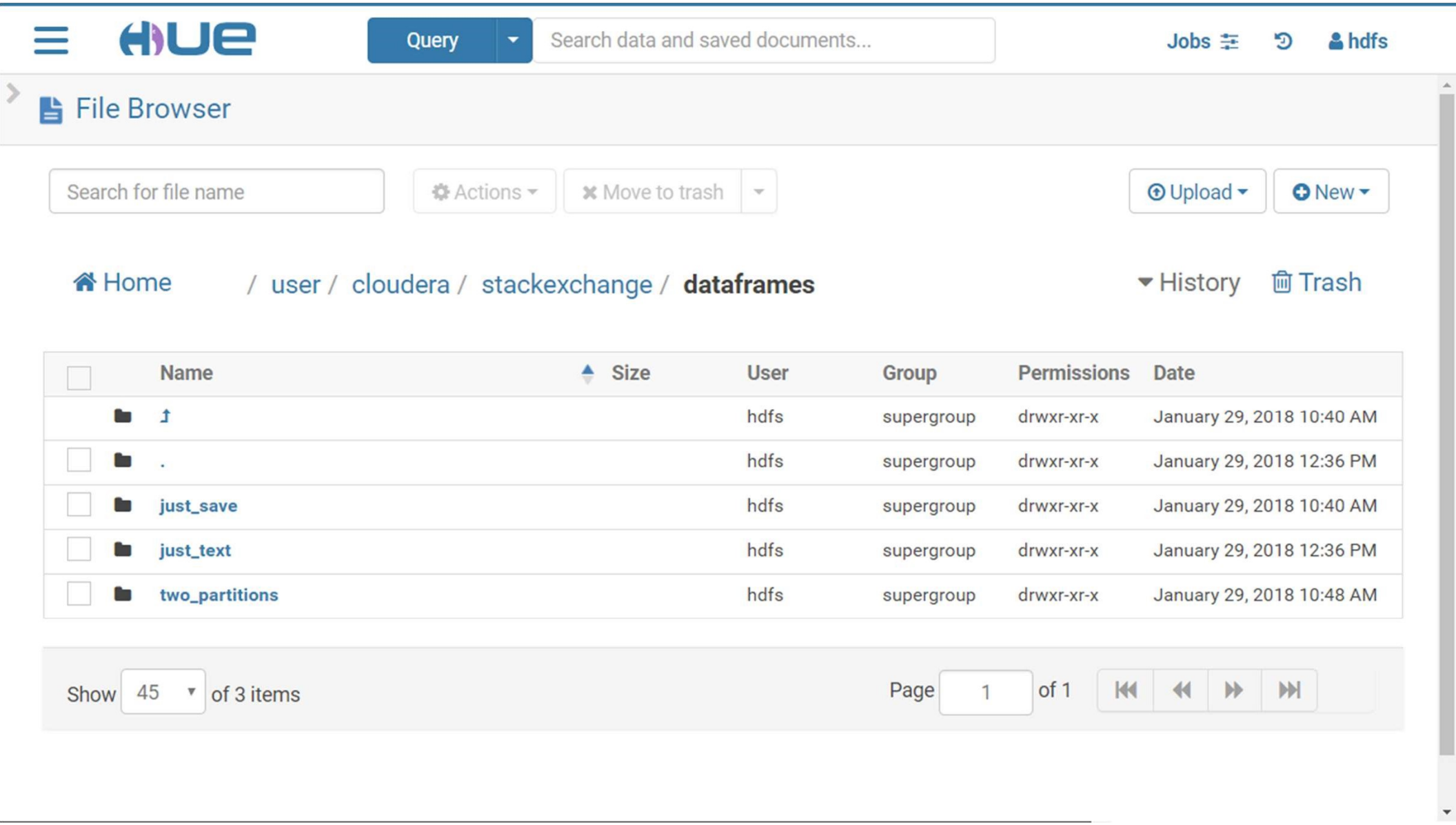
```
postsDF.write.format("text").save("/user/cloudera/
  stackexchange/dataframes/just_text")

postsDF.select("Title").write.format("text").save("/user/
  cloudera/stackexchange/dataframes/just_text_title")
```

Save as Text

Use format()

Similar to RDDs

- Almost... multiple columns not supported
- Possible to save single column


HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange / dataframes
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
. hdfs supergroup drwxr-xr-x January 29, 2018 12:36 PM
just_save hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
just_text hdfs supergroup drwxr-xr-x January 29, 2018 12:36 PM
two_partitions hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
Show 45 of 3 items
Page 1 of 1

Query | Search data and saved documents... | Jobs | hdfs

File Browser

View as binary

Edit file

Download

View file location

Refresh

Last modified
01/29/2018 8:36 PM

User
hdfs

Group
supergroup

Size
245.97 KB

Mode

Home

Page 1 to 50 of 62

/ user / cloudera / stackexchange / dataframes / just_text /
part-00000-fe30a74d-5249-4b8e-956c-8f79610017b2-c000.txt

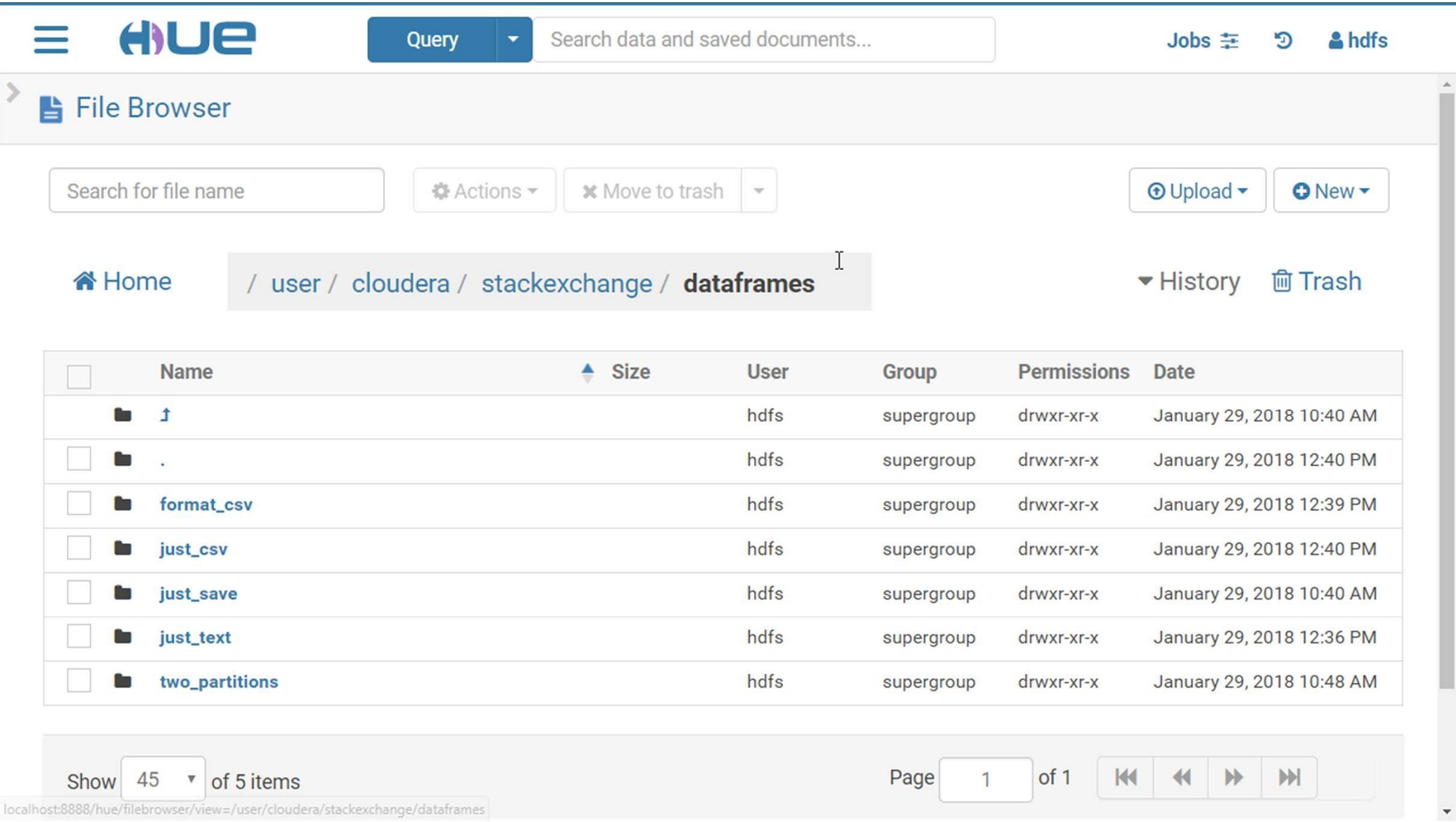
```
How can I add line numbers to Vim?

How can I show relative line numbers?
How can I change the default indentation based on filetype?

How can I use the undofile?


Can I script Vim using Python?


How can I generate a list of sequential numbers, one per line?
Does any solution exist to use vim from touch screen?


Can I use some file-tree selector which exists on graphical IDEs?
```

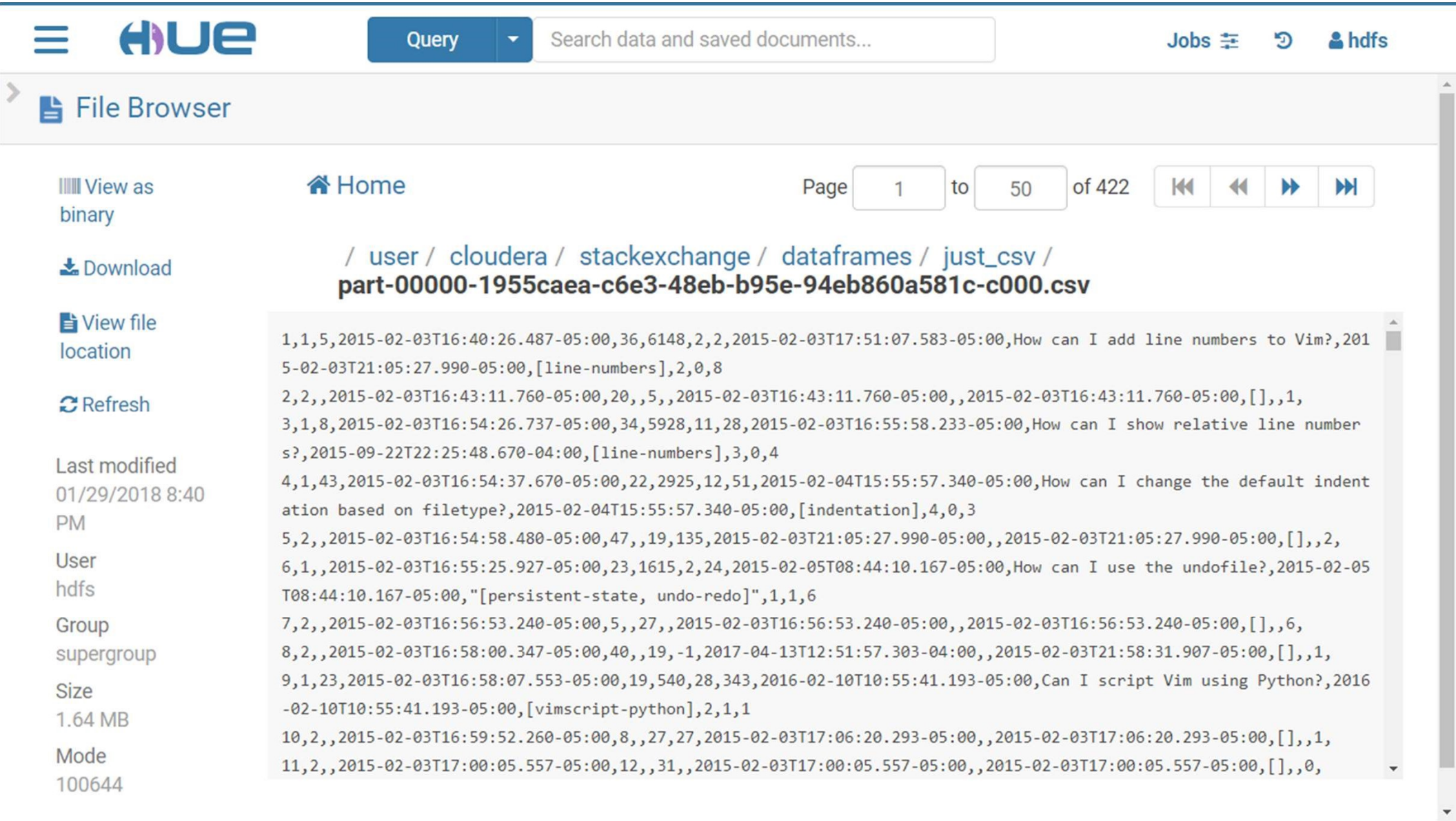
```
postsDF.write.format("csv").save("/user/cloudera/
  stackexchange/dataframes/format_csv")

postsDF.write.csv("/user/cloudera/stackexchange/dataframes/
  just_csv")
```

Save as CSV and Other Formats

Specify CSV format

- But not the only way
- You can use csv(), it is a nicer notation
- Works with other formats, try json()


HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home
/ user / cloudera / stackexchange / dataframes
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
. hdfs supergroup drwxr-xr-x January 29, 2018 12:40 PM
format_csv hdfs supergroup drwxr-xr-x January 29, 2018 12:39 PM
just_csv hdfs supergroup drwxr-xr-x January 29, 2018 12:40 PM
just_save hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
just_text hdfs supergroup drwxr-xr-x January 29, 2018 12:36 PM
two_partitions hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM
Show 45 of 5 items
Page 1 of 1
localhost:8888/hue/filebrowser/view=/user/cloudera/stackexchange/dataframes


HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
View as binary
Download
View file location
Refresh
Last modified
01/29/2018 8:40 PM
User
hdfs
Group
supergroup
Size
1.64 MB
Mode
100644
Home
Page 1 to 50 of 422
/ user / cloudera / stackexchange / dataframes / just_csv /
part-00000-1955caea-c6e3-48eb-b95e-94eb860a581c-c000.csv
1,1,5,2015-02-03T16:40:26.487-05:00,36,6148,2,2,2015-02-03T17:51:07.583-05:00,How can I add line numbers to Vim?,2015-02-03T21:05:27.990-05:00,[line-numbers],2,0,8
2,2,,2015-02-03T16:43:11.760-05:00,20,,5,,2015-02-03T16:43:11.760-05:00,,2015-02-03T16:43:11.760-05:00,[],,1,
3,1,8,2015-02-03T16:54:26.737-05:00,34,5928,11,28,2015-02-03T16:55:58.233-05:00,How can I show relative line numbers?,2015-09-22T22:25:48.670-04:00,[line-numbers],3,0,4
4,1,43,2015-02-03T16:54:37.670-05:00,22,2925,12,51,2015-02-04T15:55:57.340-05:00,How can I change the default indentation based on filetype?,2015-02-04T15:55:57.340-05:00,[indentation],4,0,3
5,2,,2015-02-03T16:54:58.480-05:00,47,,19,135,2015-02-03T21:05:27.990-05:00,,2015-02-03T21:05:27.990-05:00,[],,2,
6,1,,2015-02-03T16:55:25.927-05:00,23,1615,2,24,2015-02-05T08:44:10.167-05:00,How can I use the undofile?,2015-02-05T08:44:10.167-05:00,"[persistent-state, undo-redo]",1,1,6
7,2,,2015-02-03T16:56:53.240-05:00,5,,27,,2015-02-03T16:56:53.240-05:00,,2015-02-03T16:56:53.240-05:00,[],,6,
8,2,,2015-02-03T16:58:00.347-05:00,40,,19,-1,2017-04-13T12:51:57.303-04:00,,2015-02-03T21:58:31.907-05:00,[],,1,
9,1,23,2015-02-03T16:58:07.553-05:00,19,540,28,343,2016-02-10T10:55:41.193-05:00,Can I script Vim using Python?,2016-02-10T10:55:41.193-05:00,[vimscript-python],2,1,1
10,2,,2015-02-03T16:59:52.260-05:00,8,,27,27,2015-02-03T17:06:20.293-05:00,,2015-02-03T17:06:20.293-05:00,[],,1,
11,2,,2015-02-03T17:00:05.557-05:00,12,,31,,2015-02-03T17:00:05.557-05:00,,2015-02-03T17:00:05.557-05:00,[],,0,

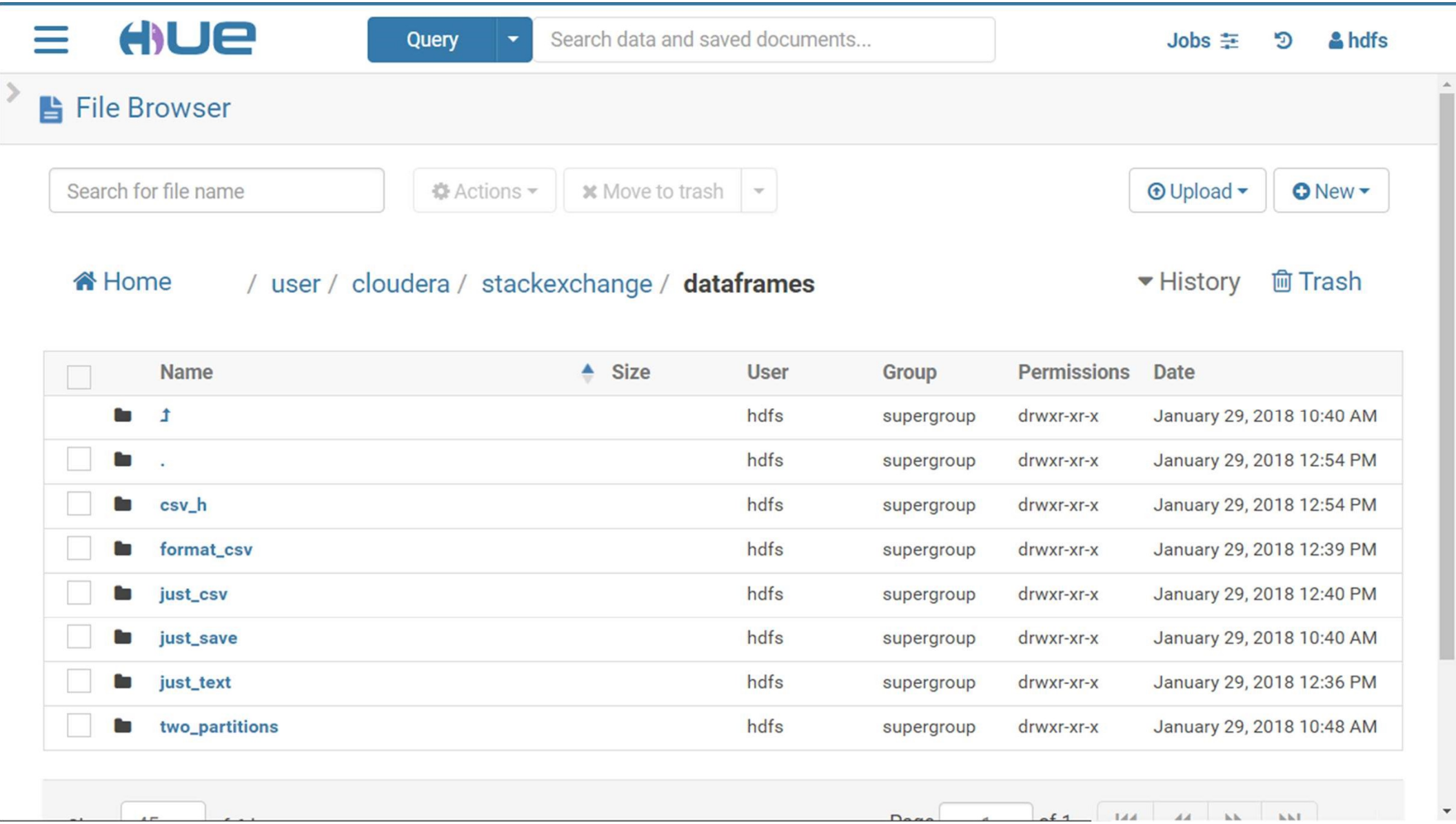
```
postsDF.write
  .option("header", "true")
  .csv("/user/cloudera/stackexchange/dataframes/csv_h")
```

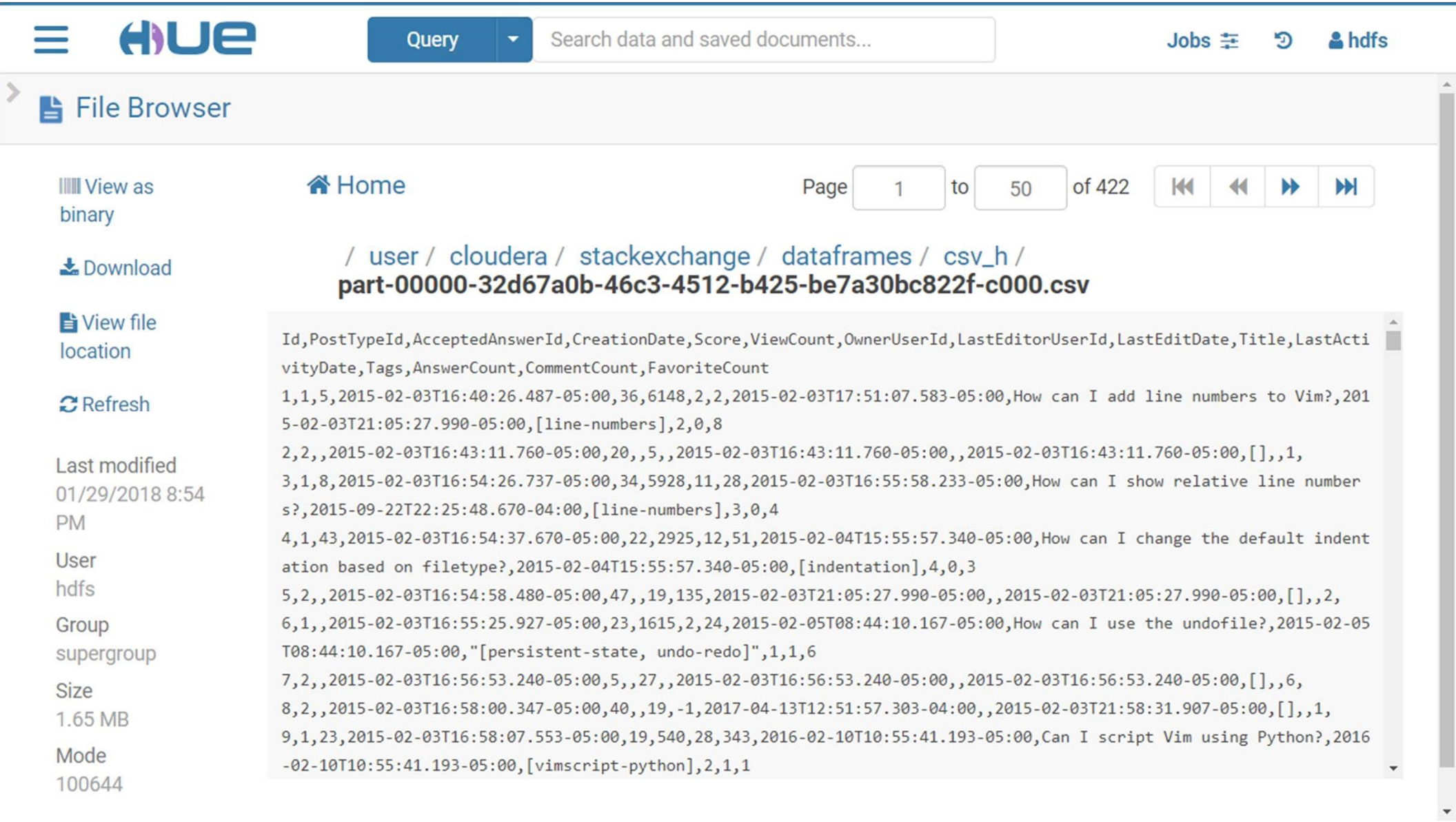
Output Options

Specify configuration parameters

Use option() or options()

- i.e. include header
- Set separator


HUE
Query
Search data and saved documents...
Jobs
hdfs
File Browser
Search for file name
Actions
Move to trash
Upload
New
Home / user / cloudera / stackexchange / dataframes
History
Trash
Name
Size
User
Group
Permissions
Date
hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
. hdfs supergroup drwxr-xr-x January 29, 2018 12:54 PM
csv_h hdfs supergroup drwxr-xr-x January 29, 2018 12:54 PM
format_csv hdfs supergroup drwxr-xr-x January 29, 2018 12:39 PM
just_csv hdfs supergroup drwxr-xr-x January 29, 2018 12:40 PM
just_save hdfs supergroup drwxr-xr-x January 29, 2018 10:40 AM
just_text hdfs supergroup drwxr-xr-x January 29, 2018 12:36 PM
two_partitions hdfs supergroup drwxr-xr-x January 29, 2018 10:48 AM


Query
Search data and saved documents...
Jobs
hdfs
File Browser
View as binary
Download
View file location
Refresh
Last modified
01/29/2018 8:54 PM
User
hdfs
Group
supergroup
Size
1.65 MB
Mode
100644
Home
Page 1 to 50 of 422
/ user / cloudera / stackexchange / dataframes / csv_h /
part-00000-32d67a0b-46c3-4512-b425-be7a30bc822f-c000.csv
Id,PostTypeId,AcceptedAnswerId,CreationDate,Score,ViewCount,OwnerUserId,LastEditorUserId,LastEditDate,Title,LastActivityDate,Tags,AnswerCount,CommentCount,FavoriteCount
1,1,5,2015-02-03T16:40:26.487-05:00,36,6148,2,2,2015-02-03T17:51:07.583-05:00,How can I add line numbers to Vim?,2015-02-03T21:05:27.990-05:00,[line-numbers],2,0,8
2,2,,2015-02-03T16:43:11.760-05:00,20,,5,,2015-02-03T16:43:11.760-05:00,,2015-02-03T16:43:11.760-05:00,[],,1,
3,1,8,2015-02-03T16:54:26.737-05:00,34,5928,11,28,2015-02-03T16:55:58.233-05:00,How can I show relative line numbers?,2015-09-22T22:25:48.670-04:00,[line-numbers],3,0,4
4,1,43,2015-02-03T16:54:37.670-05:00,22,2925,12,51,2015-02-04T15:55:57.340-05:00,How can I change the default indentation based on filetype?,2015-02-04T15:55:57.340-05:00,[indentation],4,0,3
5,2,,2015-02-03T16:54:58.480-05:00,47,,19,135,2015-02-03T21:05:27.990-05:00,,2015-02-03T21:05:27.990-05:00,[],,2,
6,1,,2015-02-03T16:55:25.927-05:00,23,1615,2,24,2015-02-05T08:44:10.167-05:00,How can I use the undofile?,2015-02-05T08:44:10.167-05:00,"[persistent-state, undo-redo]",1,1,6
7,2,,2015-02-03T16:56:53.240-05:00,5,,27,,2015-02-03T16:56:53.240-05:00,,2015-02-03T16:56:53.240-05:00,[],,6,
8,2,,2015-02-03T16:58:00.347-05:00,40,,19,-1,2017-04-13T12:51:57.303-04:00,,2015-02-03T21:58:31.907-05:00,[],,1,
9,1,23,2015-02-03T16:58:07.553-05:00,19,540,28,343,2016-02-10T10:55:41.193-05:00,Can I script Vim using Python?,2016-02-10T10:55:41.193-05:00,[vimscript-python],2,1,1

```
postsDF.rdd.
  saveAsTextFile("/user/cloudera/stackexchange/dataframes/just_rdd")

postsDF.select('Title').write.mode('overwrite')
  .format('text').save('/user/cloudera/stackexchange/dataframes/just_df')

postsDF.write.mode("overwrite")
  .csv("/user/cloudera/stackexchange/dataframes/repeat_df")
```

SaveMode

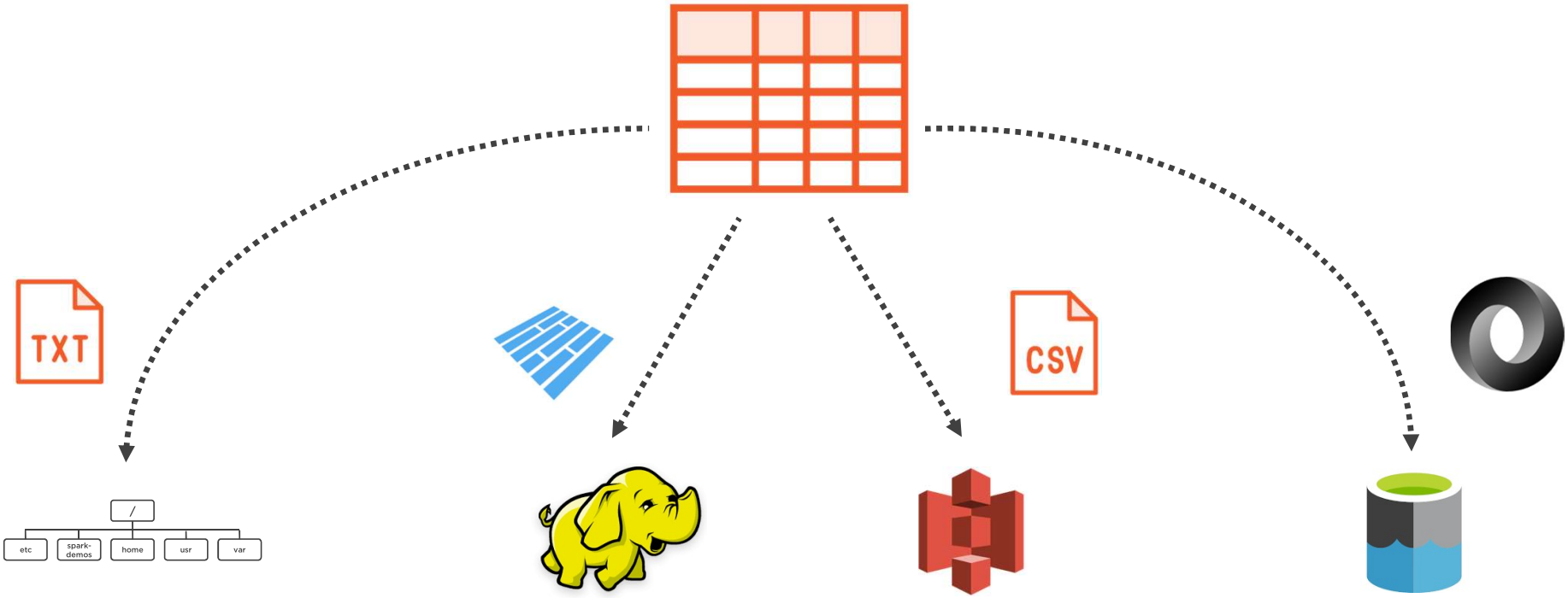
Another difference, remember this error with RDDs?

- FileAlreadyExistsException: Output directory already exists

Specify savemode, with mode()

- **Available values are append, overwrite, error, and ignore**

Saving DataFrames



Spark SQL

```
Select Id From Posts
```

```
postsDF.select("Id").show()
```

```
Select Id From Posts
```

```
postsDF.("Select Id From Posts")
.show()
```

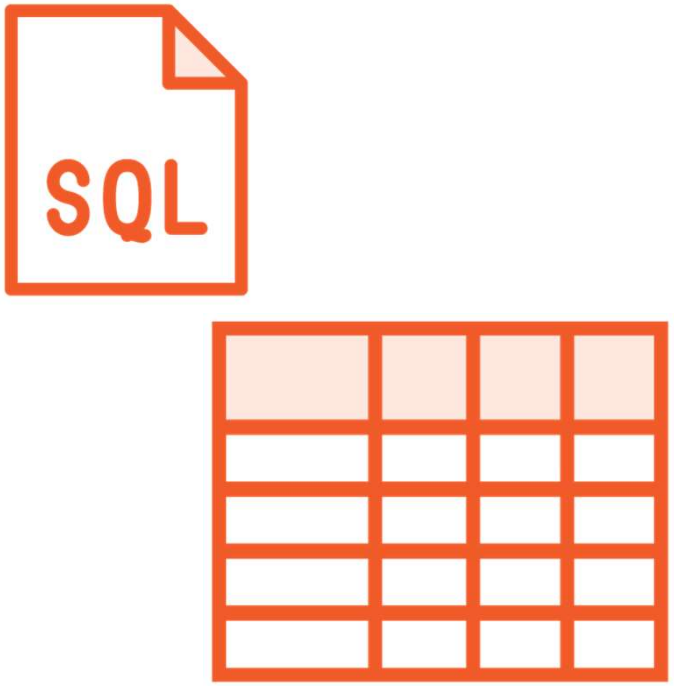
```
spark.sql("select * from postsDF").show()
```

Running SQL Queries in Spark SQL

Hello SQL my old friend...

- Use spark.sql() to execute queries
- Not so fast, you need a bit more than a DataFrame
- You need a view

Temporary Views in Spark SQL



Table

Register DataFrame as Temporary View

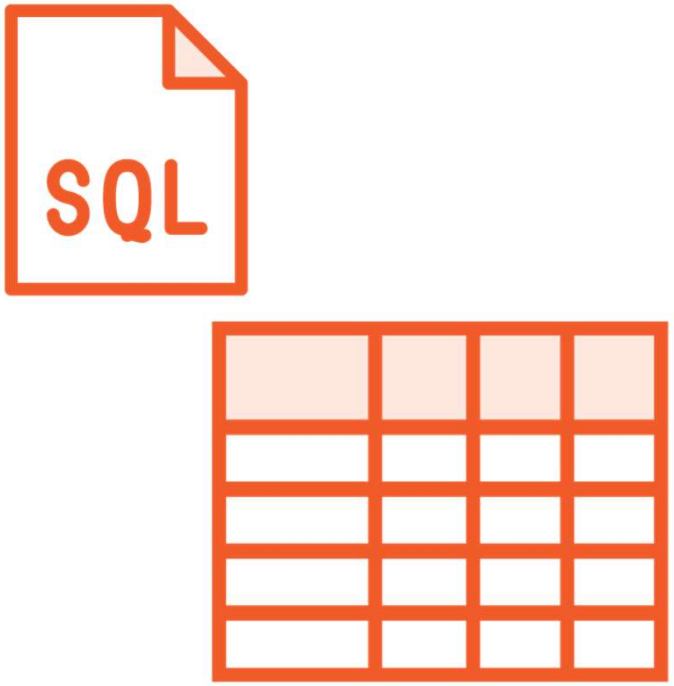
- createOrReplaceTempView()
- Formerly called registerTempTable()

Allows running SQL queries on DataFrames

- Uses .sql() method

Results returned as DataFrame

Some Details



Regarding SQL

- SQL 2003
- Native SQL parser
- Subquery support

```
postsDF.createOrReplaceTempView("Posts")

spark.sql("select count(*) from Posts")

spark.sql("select count(*) from Posts").show()

spark.sql("select count(*) as TotalPosts from Posts").show()

spark.sql("select Id, Title, ViewCount, AnswerCount from
  Posts").show(5, truncate=false)
```

Temporary Views in Spark SQL

Register our view using createOrReplaceTempView()

Query using SQL

Working directly with SQL opens up a world of possibilities

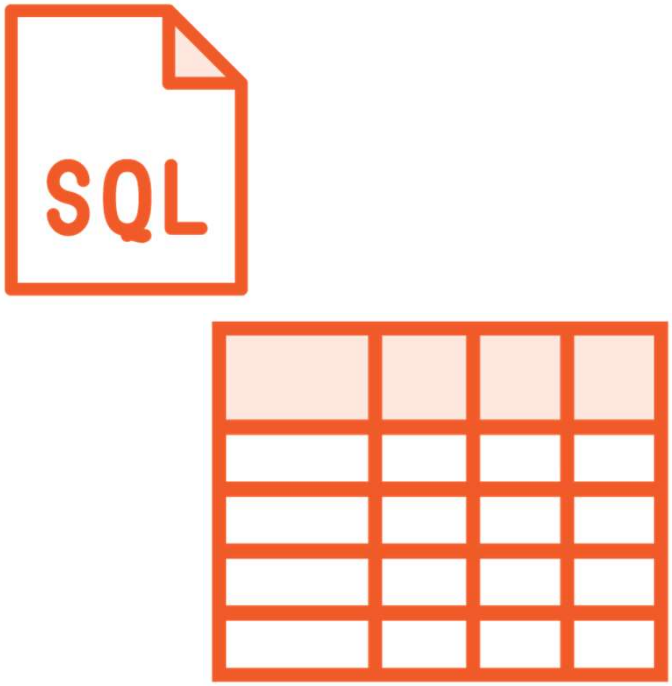
```
val top_viewsDF = spark.sql("select Id, reverse(Title) as
  eltiT, ViewCount, AnswerCount from Posts order by ViewCount
  desc")

top_viewsDF.select("eltiT", "ViewCount").show(5,
  truncate=false)
```

Spark SQL + DataFrames

Combine Spark SQL and DataFrames queries

- Return DataFrames
- And use functions

Views in Spark SQL



Session scoped

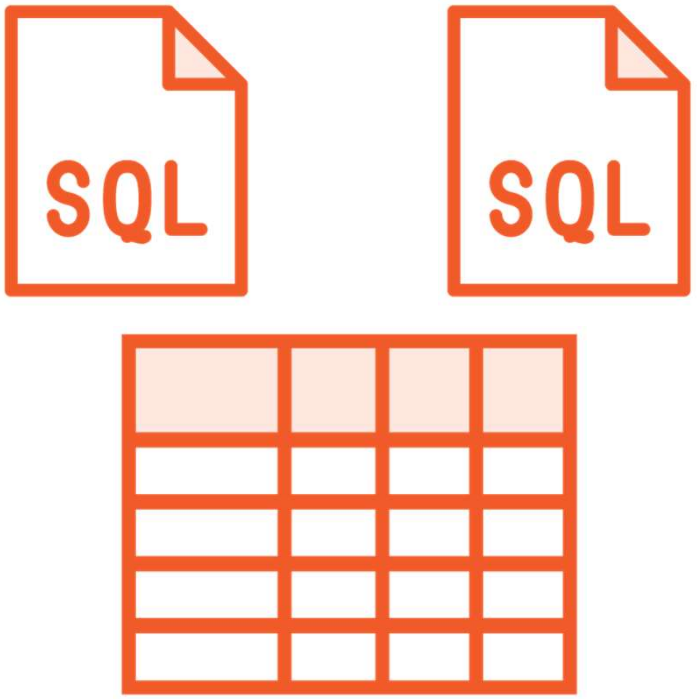
Cannot be accessed from other sessions

Dropped once session or application ends

Or explicitly

- dropTempView()

Global Views in Spark SQL



Possible to create Global Temporary Views

Cross-session

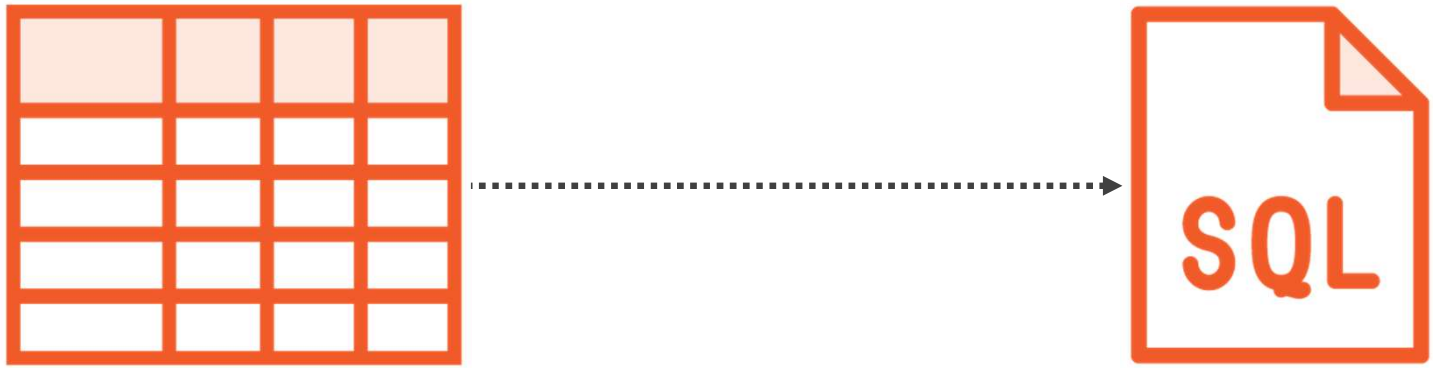
- Not dropped when session ends

Tied to global_temp

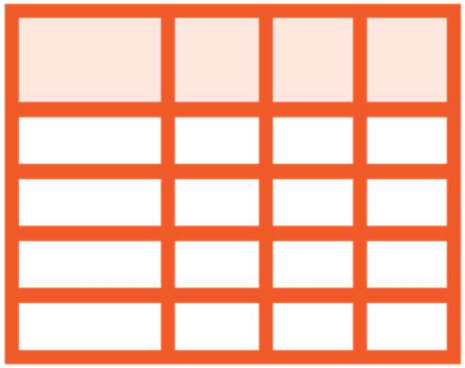
Need to use qualified name for access

Dropped once application terminates

Spark SQL Views



Loading Files Using Spark SQL



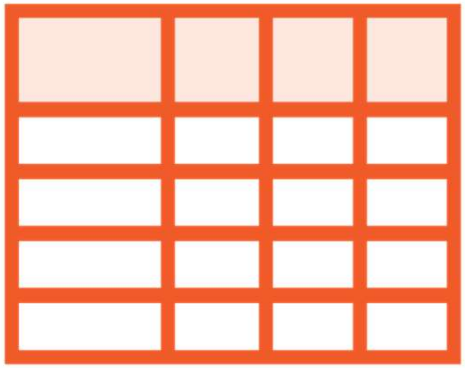
Use DataFrames

Create Temporary Views

Another useful functionality

- Directly from files with Spark SQL

Loading Files Using Spark SQL



Parquet files

- Work great because...

Schema information available

```
val comments_loadedDF = spark.sql("select * from
  parquet.`/user/cloudera/stackexchange/comments_parquet`")

comments_loadedDF.show(5)

spark.sql("select * from parquet.`/user/cloudera/
  stackexchange/comments_parquet` order by Score
  desc").show(5)
```

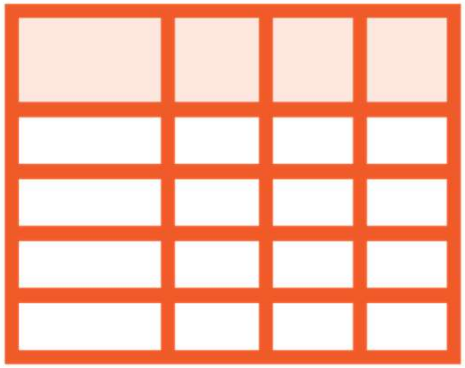
SQL Directly on Parquet Files

Use parquet.``

Load from file into DataFrame

Query normally

Loading View as DataFrame



Load Temporary View into DataFrame

Using table()

Query using DSL

```
comments_loadedDF.createOrReplaceTempView("comments")
val comments_reloadedDF = spark.table("comments")
comments_reloadedDF.orderBy($"score".desc).show(5)
```

Loading View as DataFrame

Use table()

Equivalent to loading using Select *

Saving to Persistent Tables

Available after application restart

Save to a persistent table

- Hive metastore

Spark can create a local Hive metastore

- Derby

```
postsDF.write.saveAsTable('posts_savetable_nooption')
spark.sql('select * from posts_savetable_nooption').show()
```

Save to Persistent Tables

Persist using saveAsTable()

Use spark.sql() or table() for querying

Something went wrong... What can it be?

Spark 2 Known Issues

The following sections describe the current known issues and limitations in Cloudera Distribution of Apache Spark 2. In some cases, a feature from the upstream Apache Spark project is currently not considered reliable enough to be supported by Cloudera. For a number of integration features in CDH that rely on Spark, the feature does not work with Cloudera Distribution of Apache Spark 2 because CDH components are not introducing dependencies on Spark 2.

Continue reading:

Empty result when reading Parquet table created by saveAsTable()

After a Parquet table is created by the `saveAsTable()` function, Spark SQL queries against the table return an empty result set. The issue is caused by the "path" property of the table not being written to the Hive metastore during the `saveAsTable()` call.

Bug: SPARK-21994

Affects: Cloudera Distribution of Apache Spark 2.1 release 2, Cloudera Distribution of Apache Spark 2.2 release 1

Severity: High

Workaround: You can set the path manually before the call to `saveAsTable()`:

```
val options = Map("path" -> "/path/to/hdfs/directory/containing/table")
df.write.options(options).saveAsTable("db_name.table_name")
```

Or you can add the path to the metastore when the table already exists, for example:

```
spark.sql("alter table db_name.table_name set SERDEPROPERTIES ('path'='hdfs://host.example.com:8020/ware
spark.catalog.refreshTable("db_name.table_name")
```

```
postsDF.write.option('path', '/user/cloudera/stackexchange/tables')
.saveAsTable('posts_savetable_option')

spark.sql('select * from posts_savetable_option').show()
```

Save to Persistent Tables

Known issue!

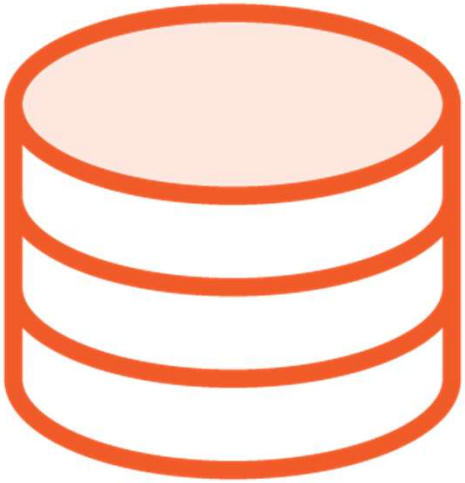
Try again

Now it works

saveAsTable()

name	Table name
format	Parquet, csv, txt, json
mode	Append: *add to existing data* Overwrite: *replace existing data* Error: *throw exception if data already exists* Ignore: *ignore this operation if data exists, no exception throw*
partitionBy	Columns used for partitioning
options	Other available options

Querying External Databases



Read/write from databases using JDBC

- Java DataBase Connectivity

Include driver in classpath

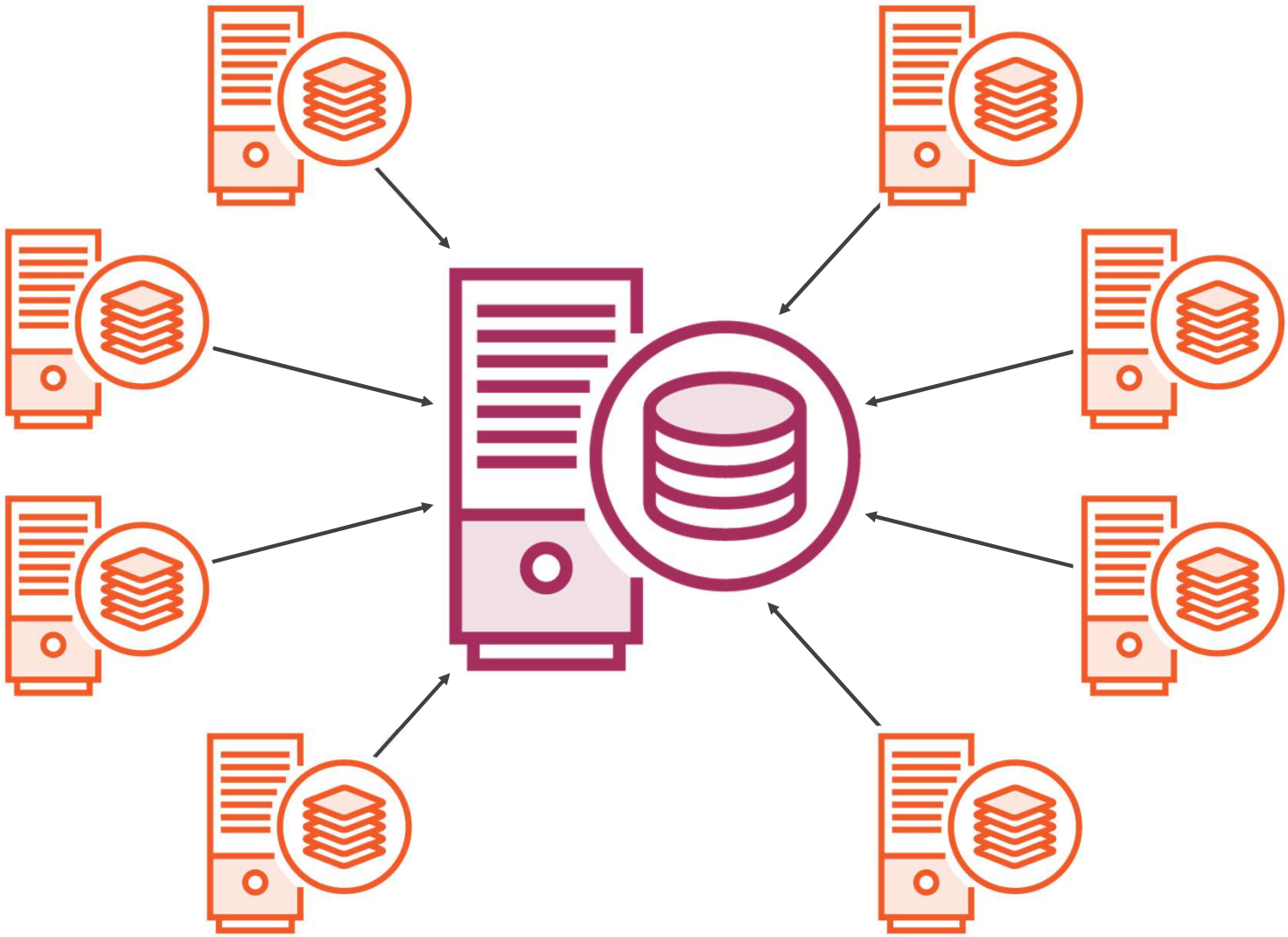
- MySQL driver in our case

Query using jdbc()

Warning!

Massively Parallel Computing

Possible to take down your own database



```
pyspark2 --jars mysql-connector-java-5.1.41-bin.jar

external_databaseDF = spark.read.format("jdbc")
     .option("url","jdbc:mysql://dn04.cloudera/scm")
     .option("driver", "com.mysql.jdbc.Driver")
     .option("dbtable", "COMMANDS")
     .option("user", "scm_user")
     .option("password", "scm_pwd")
     .load()

external_databaseDF.count()

external_databaseDF.columns

external_databaseDF.show()
```

External Databases

Specify JDBC MySQL driver when starting

Let's test with Cloudera Manager's MySQL database

- A few necessary parameters

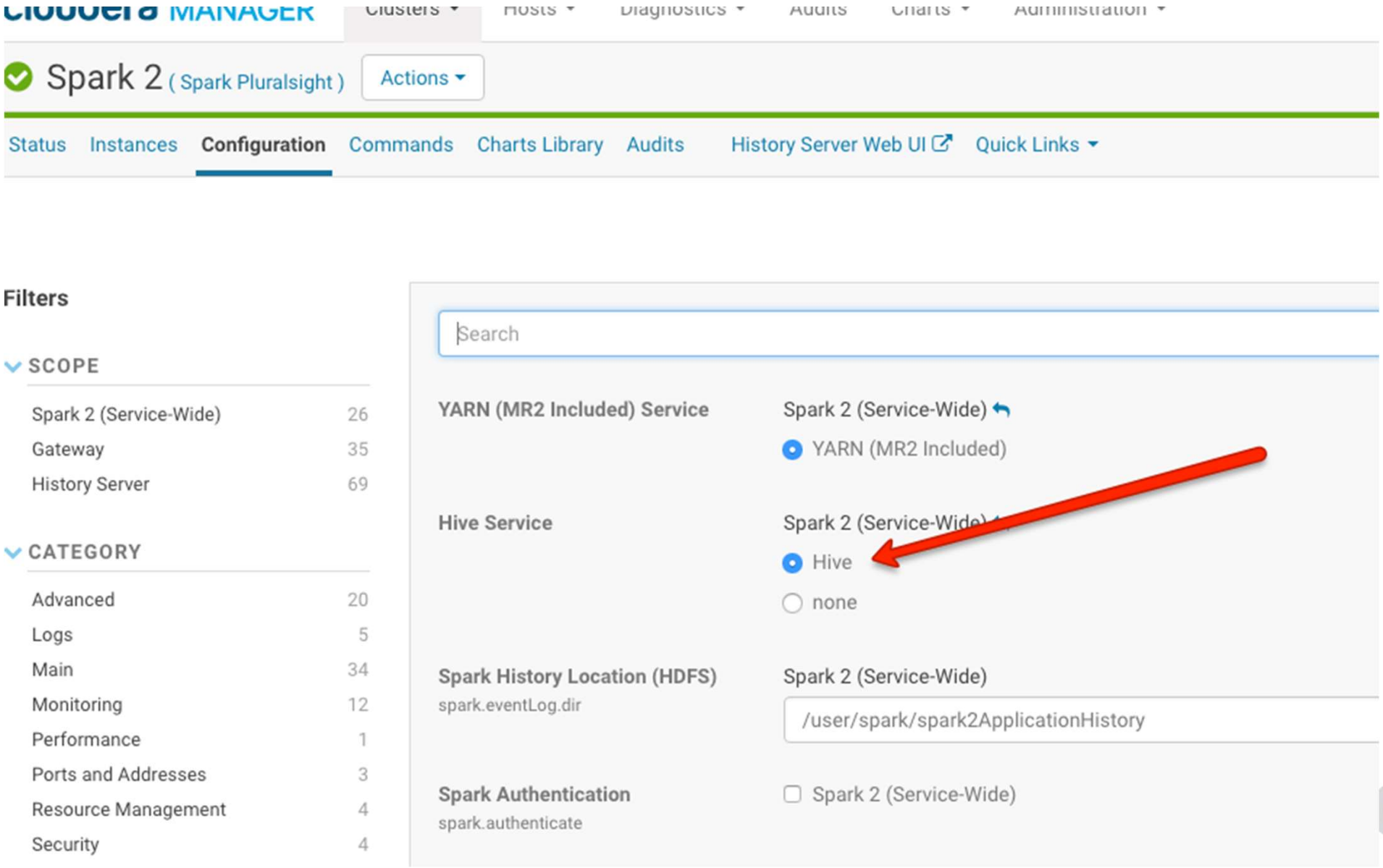
Apache Hive

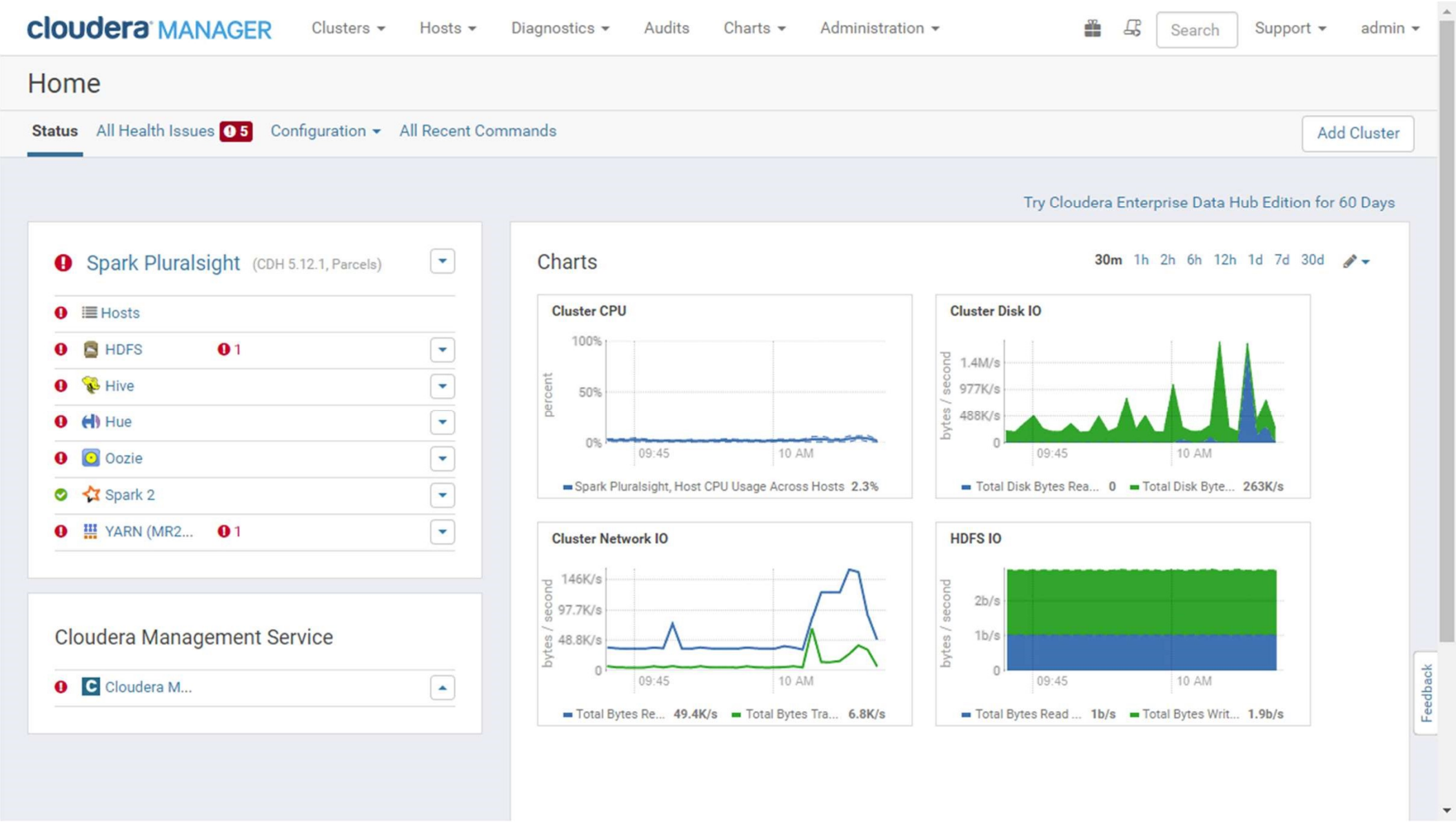
Read and write from Hive

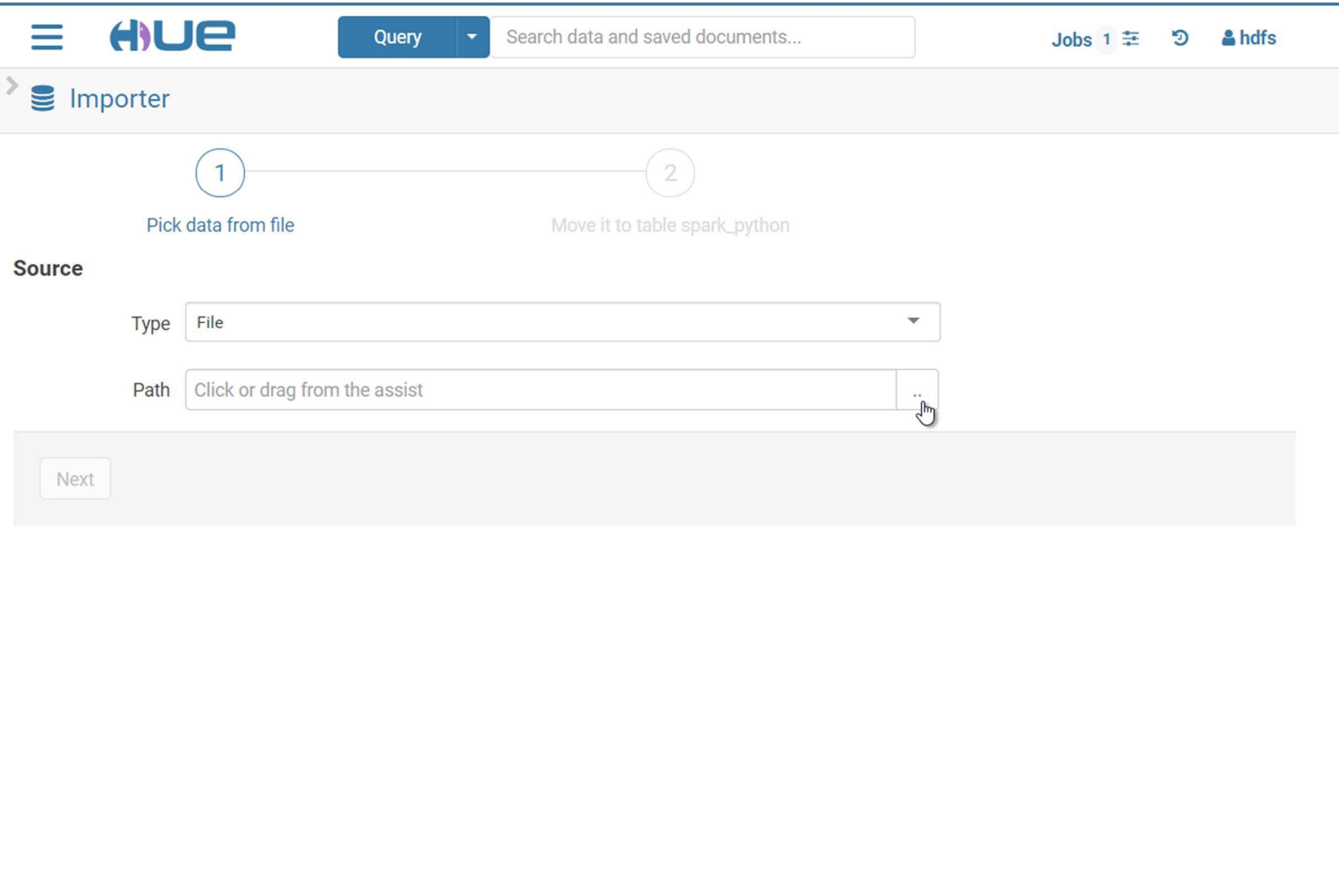
Configuration and dependencies

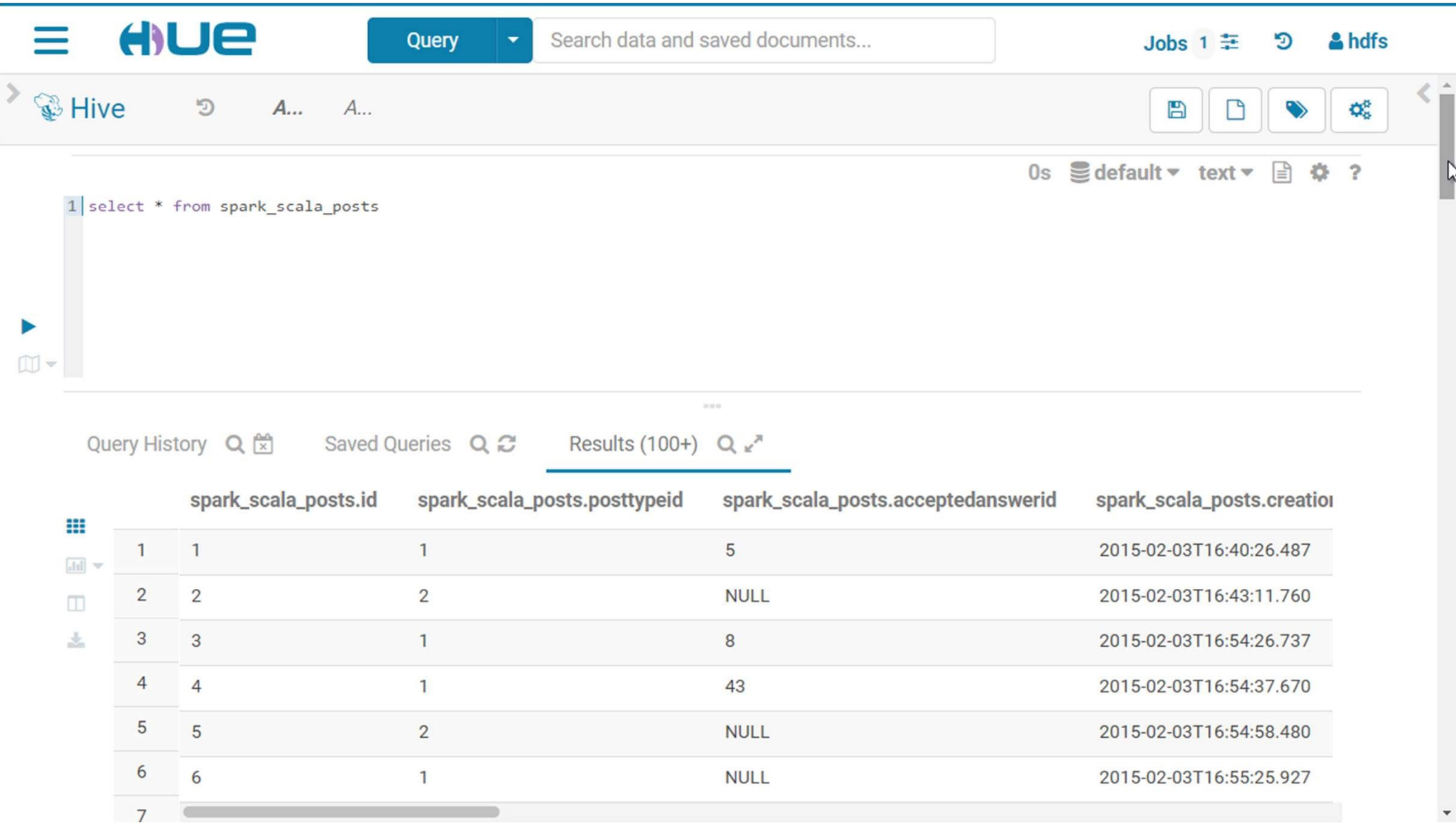
Explicitly state in session

- enableHiveSupport()


Clusters
Hosts
Diagnostics
Audits
Charts
Administration
Spark 2 (Spark Pluralsight)
Actions
Status
Instances
Configuration
Commands
Charts Library
Audits
History Server Web UI
Quick Links
Filters
SCOPE
Spark 2 (Service-Wide) 26
Gateway 35
History Server 69
CATEGORY
Advanced 20
Logs 5
Main 34
Monitoring 12
Performance 1
Ports and Addresses 3
Resource Management 4
Security 4
Search
YARN (MR2 Included) Service
Spark 2 (Service-Wide)
YARN (MR2 Included)
Hive Service
Spark 2 (Service-Wide)
Hive
none
Spark History Location (HDFS)
spark.eventLog.dir
Spark 2 (Service-Wide)
/user/spark/spark2ApplicationHistory
Spark Authentication
spark.authenticate
Spark 2 (Service-Wide)


cloudera MANAGER
Clusters
Hosts
Diagnostics
Audits
Charts
Administration
Search
Support
admin
Home
Status
All Health Issues 5
Configuration
All Recent Commands
Add Cluster
Try Cloudera Enterprise Data Hub Edition for 60 Days
Spark Pluralsight (CDH 5.12.1, Parcels)
Hosts
HDFS 1
Hive
Hue
Oozie
Spark 2
YARN (MR2... 1
Cloudera Management Service
Cloudera M...
Charts
30m 1h 2h 6h 12h 1d 7d 30d
Cluster CPU
100%
50%
0%
percent
09:45
10 AM
Spark Pluralsight, Host CPU Usage Across Hosts 2.3%
Cluster Disk IO
1.4M/s
977K/s
488K/s
0
bytes / second
09:45
10 AM
Total Disk Bytes Rea... 0
Total Disk Byte... 263K/s
Cluster Network IO
146K/s
97.7K/s
48.8K/s
0
bytes / second
09:45
10 AM
Total Bytes Re... 49.4K/s
Total Bytes Tra... 6.8K/s
HDFS IO
2b/s
1b/s
0
bytes / second
09:45
10 AM
Total Bytes Read ... 1b/s
Total Bytes Writ... 1.9b/s
Feedback


Query
Search data and saved documents...
Jobs 1
hdfs
Importer
1
Pick data from file
2
Move it to table spark_python
Source
Type
File
Path
Click or drag from the assist
..
Next


HUE
Query
Search data and saved documents...
Jobs 1
hdfs
Hive
0s
default
text
1 select * from spark_scala_posts
Query History
Saved Queries
Results (100+)
spark_scala_posts.id
spark_scala_posts.posttypeid
spark_scala_posts.acceptedanswerid
spark_scala_posts.creatio
1 1 1 5 2015-02-03T16:40:26.487
2 2 2 NULL 2015-02-03T16:43:11.760
3 3 1 8 2015-02-03T16:54:26.737
4 4 1 43 2015-02-03T16:54:37.670
5 5 2 NULL 2015-02-03T16:54:58.480
6 6 1 NULL 2015-02-03T16:55:25.927
7

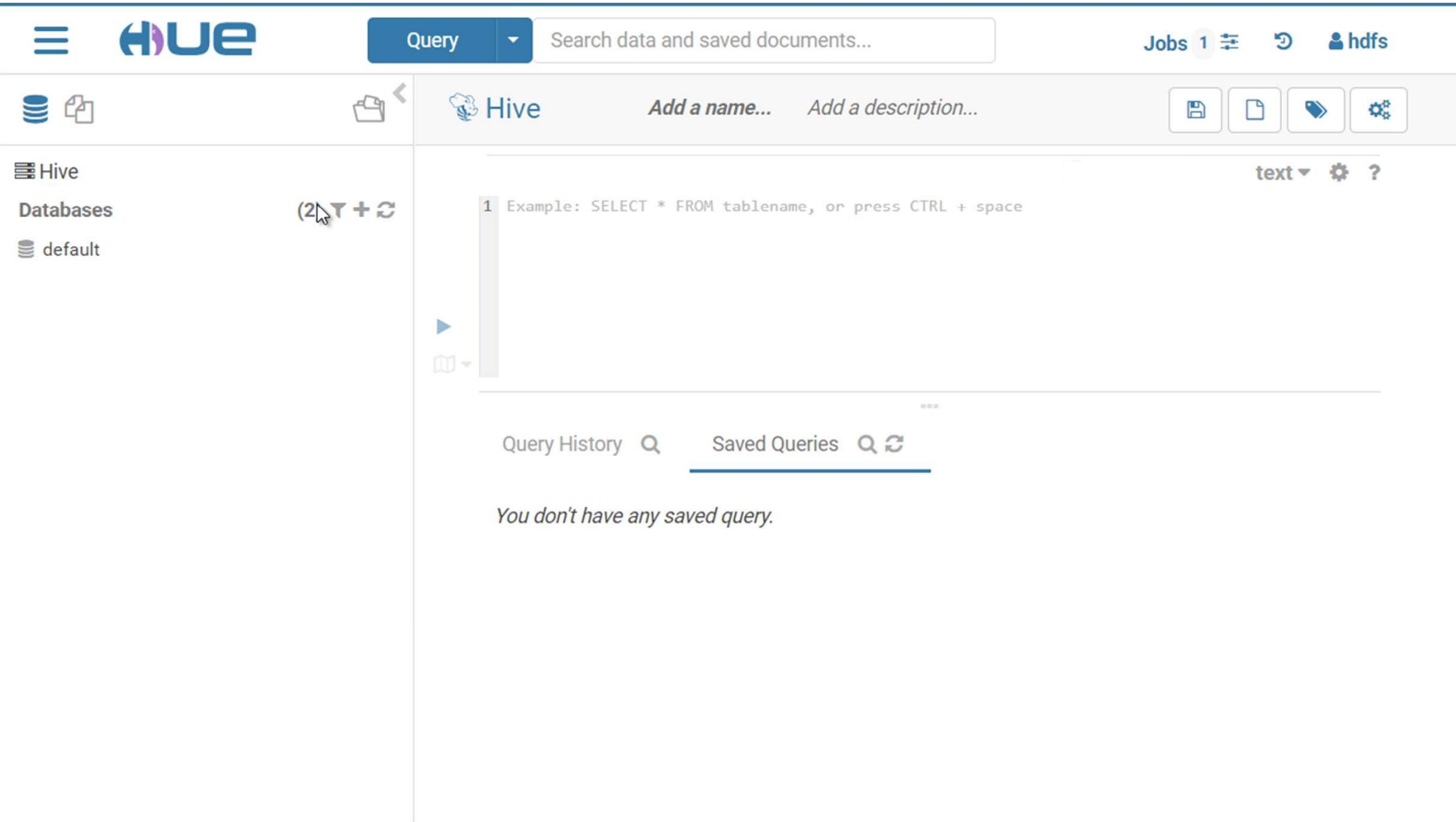
```
from os.path import abspath

warehouse_location=abspath('/user/hive/warehouses/')

spark_hive=SparkSession.builder.appName("Hive Demo with Spark").\
    config("spark.sql.warehouse.dir",warehouse_location).\
    enableHiveSupport().\
    getOrCreate()

spark_hive.sql("show databases").show()

spark_hive.sql("Select * from default.posts").show()
```

Hive

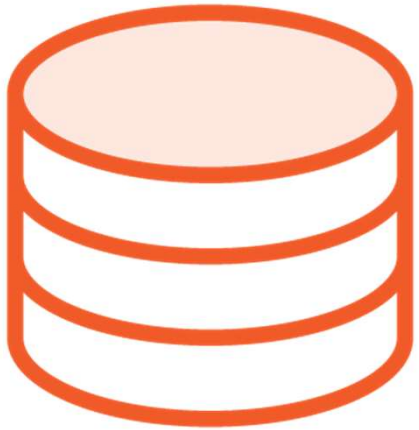
Spark SQL supports Hive formats, UDFs and metastore

- Not enabled by default

Configure and enable explicitly with enableHiveSupport()


HUE
Query
Search data and saved documents...
Jobs 1
hdfs
Hive
Add a name...
Add a description...
Hive
Databases
(2
default
text
1 Example: SELECT * FROM tablename, or press CTRL + space
Query History
Saved Queries
You don't have any saved query.

Querying Hive & External Databases



Aggregating, Grouping & Joining

Statistical analysis

- Count, or count distinct
- Group data
- Intersection of two data sets

```
val commentsDF = spark.read.parquet("/user/cloudera/
  stackexchange/comments_parquet")

commentsDF.show(5)

commentsDF.groupBy("UserId")

commentsDF.groupBy("UserId").getClass
```

Grouped Data

Use groupBy()

Returns RelationalGroupedDataset

What's available?

```
commentsDF.groupBy("UserId").sum("Score").show(5)

commentsDF.groupBy("UserId").sum("Score").sort($"sum(Score)"
  .desc).show(5)


commentsDF.groupBy("UserId").avg("Score").sort($"avg(Score)"
  .desc).show(5)
```

Explore Available Functionality

Use sum(), min(), max(), mean(). ...

Can use agg

```
commentsDF.groupBy("UserId").avg("Score").sort($"avg(Score)".desc).show(5)

commentsDF.groupBy("UserId").agg(avg("Score"),
count("Score").alias("Total")).show(5)

commentsDF.groupBy("UserId").agg(avg("Score"),
count("Score").alias("Total")).sort($"avg(Score)".desc).show(5)

commentsDF.groupBy("UserId").agg(avg("Score"),
count("Score").alias("Total")).sort($"Total".desc).show(5)
```

Multiple Aggregations

Use agg()

Sort by any column

```
commentsDF.describe().show(5)
commentsDF.describe("Score").show(5)
```

Describe

Shows column statistics

Specify columns

```
postsDF

val ask_questionsDF = postsDF.where("PostTypeId =
1").select("OwnerUserId").distinct()

val answer_questionsDF = postsDF.where("PostTypeId = 2")
  .select("OwnerUserId").distinct()
postsDF.select("OwnerUserId").distinct().count()

ask_questionsDF.count()

answer_questionsDF.count()

ask_questionsDF.join(answer_questionsDF, ask_questionsDF("OwnerUserId") ===
  answer_questionsDF("OwnerUserId")).count()

val ask = ask_questionsDF.as("ask")

val answer = answer_questionsDF.as("answer")

ask.join(answer, col("ask.OwnerUserId") === col("answer.OwnerUserId")).count()
```

```
ask_questionsDF
  .join(answer_questionsDF,
  ask_questionsDF("OwnerUserId") === answer_questionsDF("OwnerUserId"))
  .count()
```

Joining Data

Set up questionsDF and answersDF

Compare total users, those who ask and those who answer

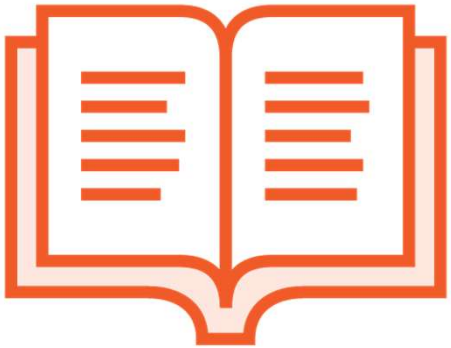
```
commentsDF.createOrReplaceTempView('comments')

spark.sql('Select UserId, count(Score) as TotalCount From comments Group
by UserId Order by  TotalCount desc').show(5)
```

Spark SQL

Besides DataFrame DSL

Available in Spark SQL

The Catalog API



Structured data

Registered views, tables, UDFs, ...

Managing metadata

Catalog API

- Interface

```
spark.catalog
spark.catalog.listDatabases()
spark.catalog.listDatabases().show()
spark.catalog.listTables("default")
spark.catalog.listTables("default").show()
spark.catalog.listTables("default").count()
spark.catalog.dropTempView("comments")
spark.catalog.listTables("default").count()
```

Catalog API

Access with spark.catalog

Can list databases, tables, functions, ...

Create or drop views among other available functionality

Takeaway

DataFrame DSL

- Querying, Filtering and Sorting
- Missing and corrupt data

Save DataFrames

- Options

Takeaway

Spark SQL

- Temporary Views
- Loading directly
- Persistent tables + known issue

Hive support

External databases

Aggregating, grouping and joining

Catalog API

Datasets