

SoSe 2014: M-TANI: Big Data Analytics

Lecture 2 – 30/04/2014

Sead Izberovic

Dr. Nikolaos Korfiatis

What is MapReduce?

- “A programming model for large-scale distributed data processing” [1]
- Inspired by the *map* and *reduce* primitives from functional programming languages [2]
- Used by:
 - Google
 - Yahoo!
 - Facebook
 - ...

Map in functional programming

- *map* takes following arguments:
 - A function **func**
 - A set of values **val**
- *map* returns the result of the computation as a set **results**

map(**func**, **val**) → **results**

Map example in Python

```
>> map(len,['Hallo', 'Big', 'Data', '!'])  
[5,3,4,1]
```

Reduce in functional programming

- *reduce* takes following arguments:
 - A binary function **bi_func**
 - A set of values **red_val**
- *reduce* returns the value of the computation **result**

reduce(**bi_func**, **red_val**) → **result**

Reduce example in Python

```
>> reduce(+, [5,3,4,1])
```

```
13
```

MapReduce execution overview

Input Data

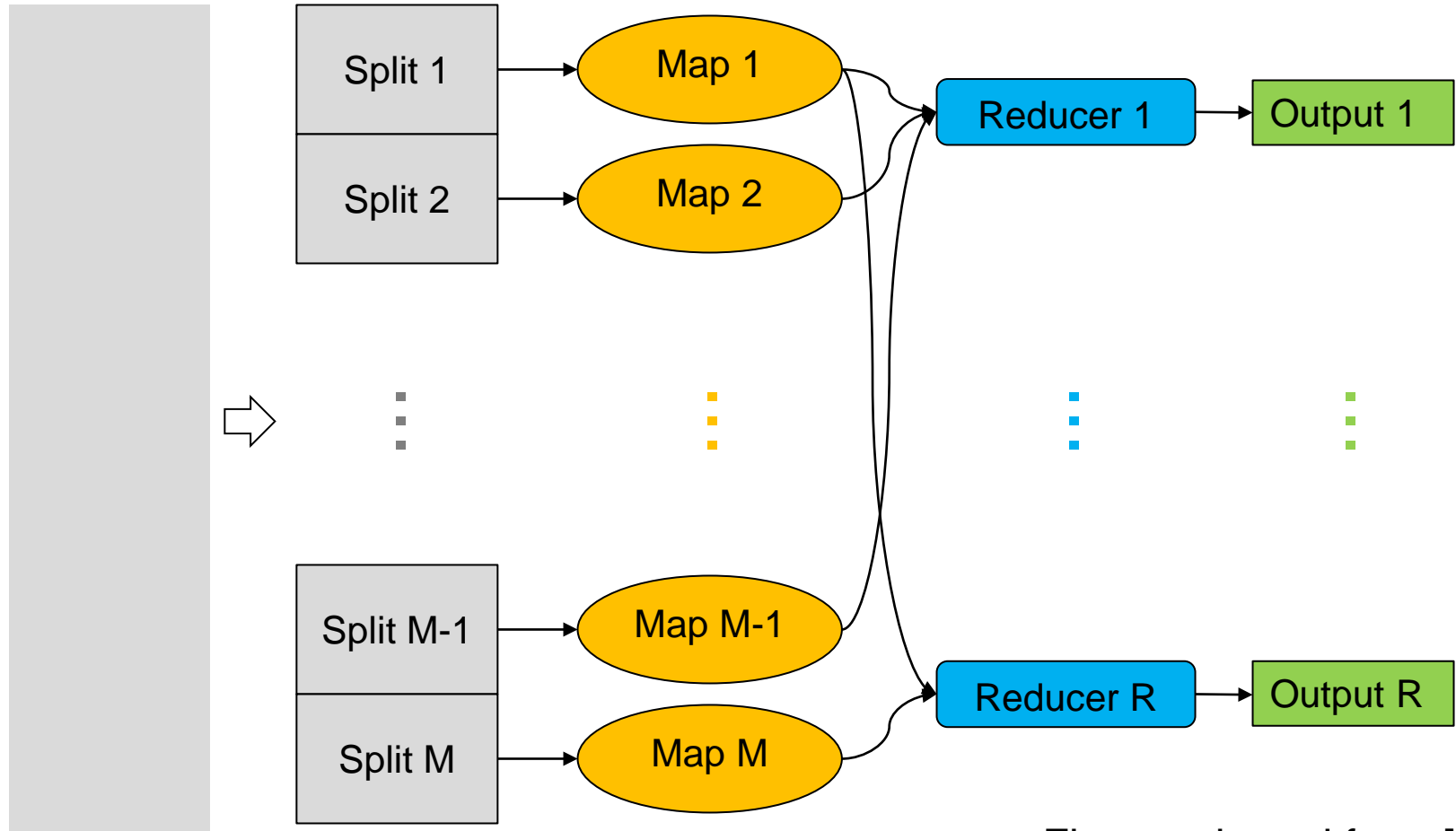


Figure adapted from [2]

Map in MapReduce

- *map* takes following arguments:
 - A key **K**
 - A value **V**
- *map* creates a set of intermediate key-values pairs

$$\text{map}(\mathbf{K}, \mathbf{V}) \rightarrow [(\mathbf{K}_1, \mathbf{V}_1), \dots, (\mathbf{K}_n, \mathbf{V}_n)]$$

Reduce in MapReduce

- *reduce* takes following arguments:
 - A key \mathbf{K}_{red}
 - A set of values $[\mathbf{V}]$
- *map* creates a set of intermediate key-values pairs

$$\text{map}(\mathbf{K}_{red}, [\mathbf{V}]) \rightarrow (\mathbf{K}_{red}, \mathbf{V}_{red})$$

MapReduce example

Word count

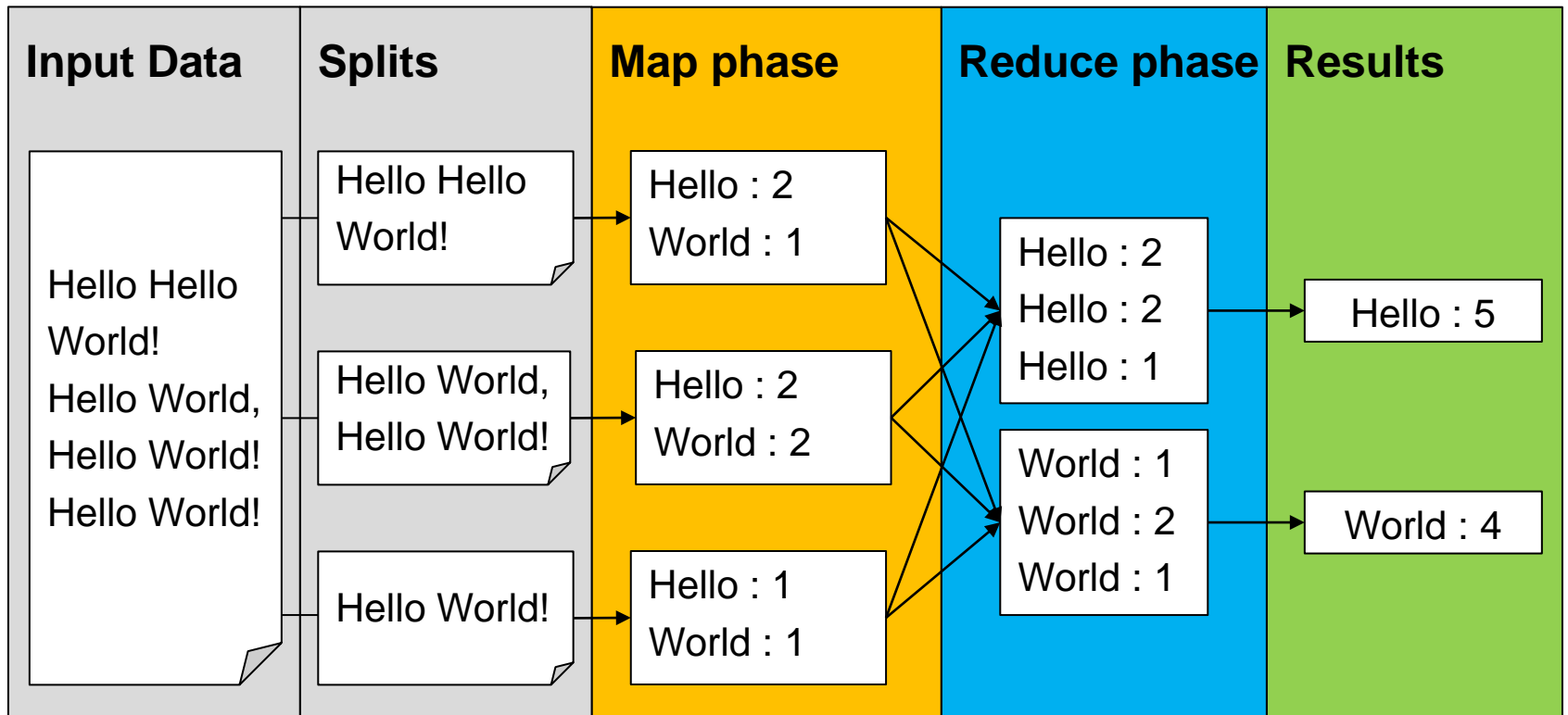


Figure adapted from [3]

Word count example in Hadoop

- Writing the Mapper

```
public static class Map extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws
    IOException {
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);
        while (tokenizer.hasMoreTokens()) {
            word.set(tokenizer.nextToken());
            output.collect(word, one);
        }
    }
}
```

source code from [4]

Word count example in Hadoop

- Writing the Reducer

```
public static class Reduce extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {
    public void reduce(Text key, Iterator<IntWritable> values, OutputCollector<Text, IntWritable> output, Reporter
reporter) throws IOException {
        int sum = 0;
        while (values.hasNext()) {
            sum += values.next().get();
        }
        output.collect(key, new IntWritable(sum));
    }
}
```

source code from [4]

Word count example in Hadoop

- Create the necessary directories

```
[cloudera@localhost ~]$ hadoop fs -mkdir /user/cloudera/wordcount
```

```
[cloudera@localhost ~]$ hadoop fs -mkdir /user/cloudera/wordcount/input
```

```
[cloudera@localhost ~]$ hadoop fs -mkdir /user/cloudera/wordcount/output
```

```
[cloudera@localhost ~]$ hadoop fs -mkdir /user/cloudera/lib
```

Word count example in Hadoop

- Copying a file from the local file system to the HDFS

```
[cloudera@localhost ~]$ hadoop fs -put input_file target_directory_in_HDFS |
```

- Copying a text file from the local file system to the HDFS

```
[cloudera@localhost ~]$ hadoop -put ./the_tragedie_of_hamlet.txt /user/cloudera/wordcount/input/
```

- Copying a jar file from the local file system to the HDFS

```
[cloudera@localhost ~]$ hadoop fs -put ./wordcount.jar /user/cloudera/lib/
```

Word count example in Hadoop

- Executing the word count example

```
[cloudera@localhost big_data]$ hadoop jar ./wordcount.jar org.myorg.WordCount  
/user/cloudera/wordcount/input/ /user/cloudera/wordcount/output/out █
```

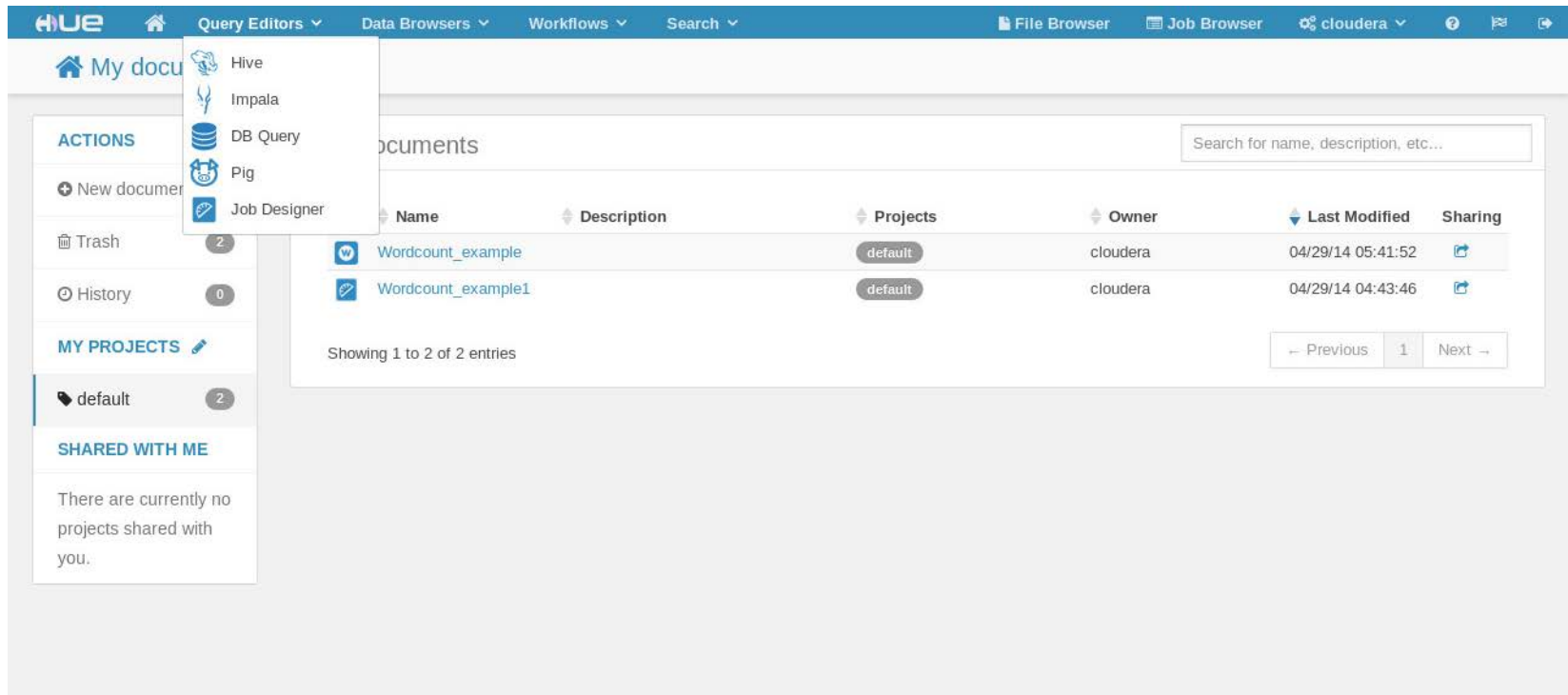
```
⋮
```

```
INFO mapreduce.Job: map 0% reduce 0%  
INFO mapreduce.Job: map 100% reduce 0%  
INFO mapreduce.Job: map 100% reduce 100%
```

```
⋮
```

Hue - Hadoop User Experience

- Web application for interacting with Hadoop



The screenshot displays the Hue web application interface. The top navigation bar includes the Hue logo and several menu items: Query Editors, Data Browsers, Workflows, Search, File Browser, Job Browser, and cloudera. The main content area shows a 'My documents' section with a search bar and a table of documents. A dropdown menu is open over the 'My documents' section, listing actions: Hive, Impala, DB Query, Pig, and Job Designer. The table below shows two documents:

Name	Description	Projects	Owner	Last Modified	Sharing
Wordcount_example		default	cloudera	04/29/14 05:41:52	
Wordcount_example1		default	cloudera	04/29/14 04:43:46	

Showing 1 to 2 of 2 entries

Navigation: Previous 1 Next

Word count example with Hue

- Creating the workflow

My Documents

Search for name, description, etc...

Name	Description	Projects	Owner	Last Modified	Sharing
Wordcount_example		default	cloudera	04/29/14 05:41:52	
Wordcount_example1		default	cloudera	04/29/14 04:43:46	

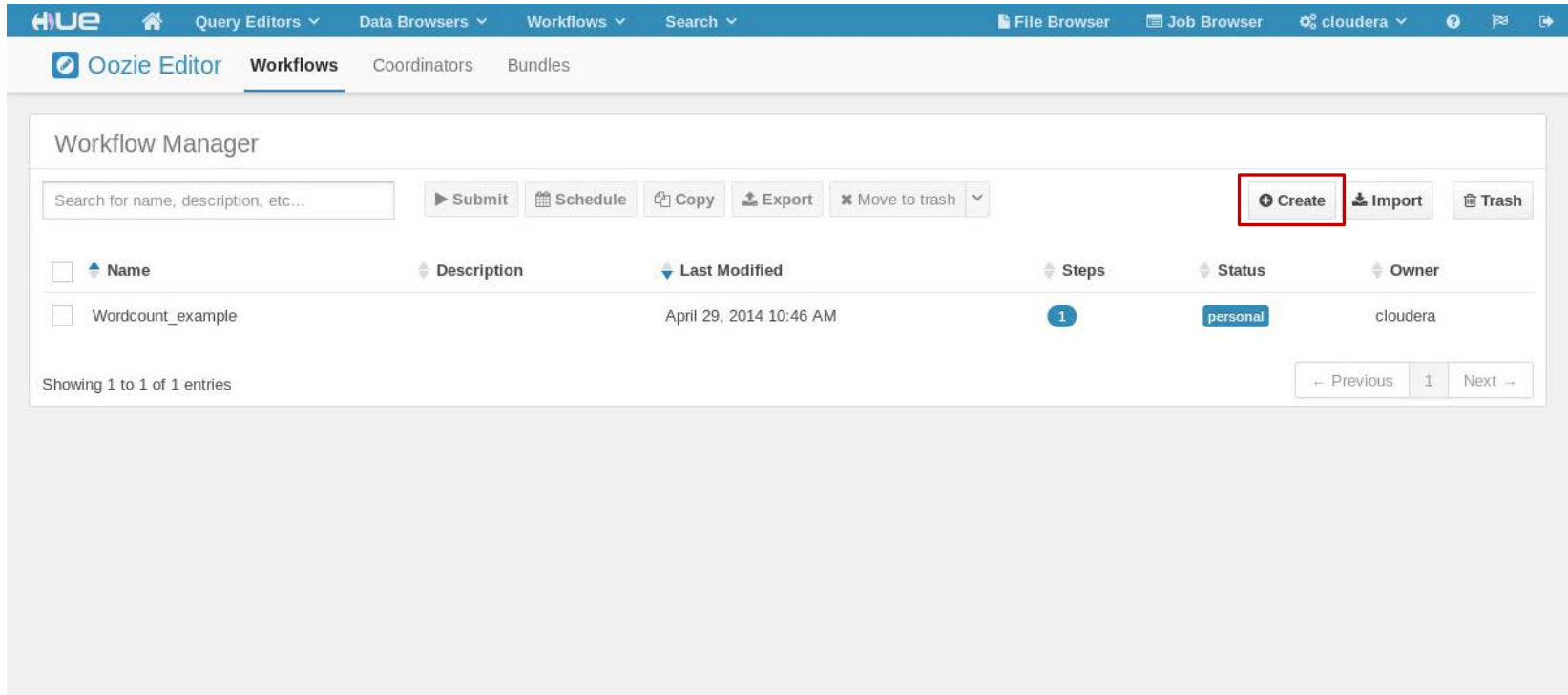
Showing 1 to 2 of 2 entries

← Previous 1 Next →

There are currently no projects shared with you.

Word count example with Hue

- Creating the workflow



The screenshot shows the Hue Oozie Editor interface. The top navigation bar includes 'HUE', 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and 'cloudera'. The main content area is titled 'Workflow Manager' and features a search bar and several action buttons: 'Submit', 'Schedule', 'Copy', 'Export', 'Move to trash', 'Create', 'Import', and 'Trash'. The 'Create' button is highlighted with a red box. Below the buttons is a table listing workflows.

<input type="checkbox"/>	Name	Description	Last Modified	Steps	Status	Owner
<input type="checkbox"/>	Wordcount_example		April 29, 2014 10:46 AM	1	personal	cloudera

Showing 1 to 1 of 1 entries

Navigation: - Previous | 1 | Next -

Word count example with Hue

- Creating the workflow

The screenshot shows the Hue Oozie Editor interface. The top navigation bar includes the Hue logo, a home icon, and dropdown menus for 'Query Editors', 'Data Browsers', 'Workflows', and 'Search'. On the right side of the navigation bar are links for 'File Browser', 'Job Browser', and 'cloudera'. Below the navigation bar, the 'Oozie Editor' section is active, with sub-tabs for 'Workflows', 'Coordinators', and 'Bundles'. On the left side, there is a 'NEW WORKFLOW' button and a 'Properties' tab. The main content area is titled 'Properties' and contains the following form fields:

- Name:
- Description:
- advanced

At the bottom of the form, there are two buttons: 'Save' and 'Back'.

Word count example with Hue

- Creating the workflow

The screenshot displays the Hue Oozie Editor interface for creating a workflow named "Word_count_example". The top navigation bar includes "HUE", "Query Editors", "Data Browsers", "Workflows", "Search", "File Browser", "Job Browser", and "cloudera". The left sidebar shows the "EDITOR" section with options for "Workflow", "Properties", "Workspace", "ADVANCED" (Import action, Kill node, History), and "ACTIONS" (Submit, Schedule). The main workspace shows a workflow diagram with a "start" node, a dashed box for an action, and an "end" node. A toolbar above the workspace contains buttons for "MapReduce", "Streaming", "Java", "Pig", "Hive", "Sqoop", "Shell", "Ssh", "DistCp", "Fs", "Email", "Sub-workflow", and "Generic". A yellow box contains the text "No actions: drag some from the panel above". An arrow points from the "Java" button to the dashed box. At the bottom, there are "Save" and "Back" buttons.

Word count example with Hue

- Creating the workflow

Edit Node: Word_count_example

Name

Description

Action type

↻ Advanced

All the paths are relative to the deployment directory. They can be absolute but this is not recommended.
You can parameterize values using case sensitive `${parameter}`.

Jar name ..

← HDFS

Main class

Cancel Done

Word count example with Hue

- Creating the workflow

Edit Node: Word_count_example

Jar name

Main class

Arguments

Java options

Capture output

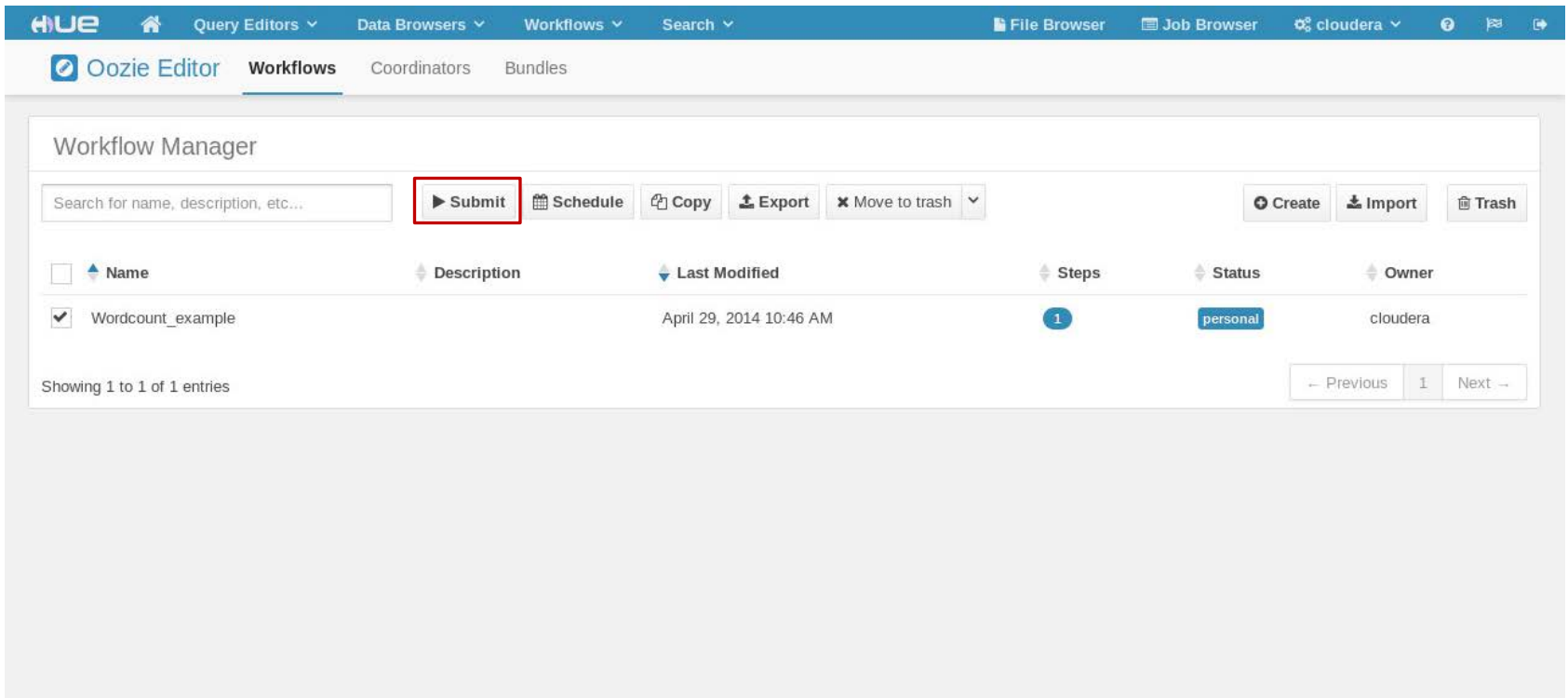
Prepare

Job properties

Files

Word count example with Hue

- Executing the word count example



The screenshot shows the Hue Oozie Editor interface. The top navigation bar includes 'HUE', 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and 'cloudera'. The main content area is titled 'Workflow Manager' and contains a search bar, a 'Submit' button (highlighted with a red box), and other action buttons like 'Schedule', 'Copy', 'Export', 'Move to trash', 'Create', 'Import', and 'Trash'. Below the search bar is a table with the following columns: Name, Description, Last Modified, Steps, Status, and Owner. The table contains one entry: 'Wordcount_example' with a last modified date of 'April 29, 2014 10:46 AM', 1 step, a 'personal' status, and an owner of 'cloudera'. The bottom of the table shows 'Showing 1 to 1 of 1 entries' and pagination controls.

Name	Description	Last Modified	Steps	Status	Owner
<input checked="" type="checkbox"/> Wordcount_example		April 29, 2014 10:46 AM	1	personal	cloudera

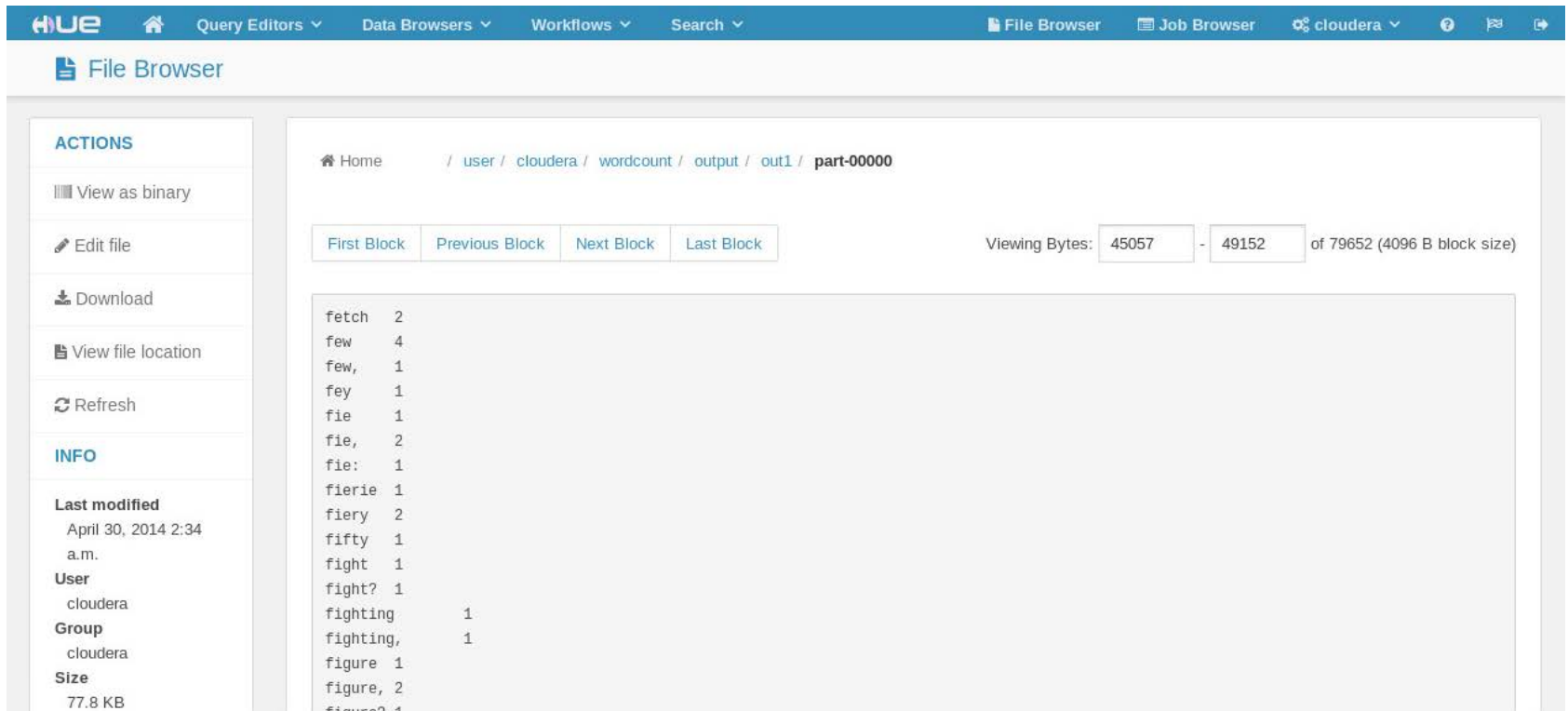
Word count example with Hue

- Executing the word count example

The screenshot displays the Hue Oozie Dashboard interface. The top navigation bar includes 'HUE', 'Query Editors', 'Data Browsers', 'Workflows', 'Search', 'File Browser', 'Job Browser', and 'cloudera'. The main content area is titled 'Oozie Dashboard' and 'Workflows'. The workflow 'Wordcount_example' is selected, and its 'Graph' view is shown. The workflow consists of three actions: 'start', 'java', and 'end', all of which are marked as 'OK'. The 'java' action is expanded to show the details of the 'Word_count_example' job, which is currently 'Counting words'. On the left sidebar, the 'STATUS' section shows 'SUCCEEDED' and the 'PROGRESS' section shows a 100% completion bar. The 'SUBMITTER' is 'cloudera' and the 'ID' is '0000025-140429033440090-oozie-oozi-W'.

Word count example with Hue

- The results



The screenshot shows the Hue File Browser interface. The breadcrumb path is: Home / user / cloudera / wordcount / output / out1 / part-00000. The file size is 77.8 KB, last modified on April 30, 2014 at 2:34 a.m. The user is 'cloudera' and the group is 'cloudera'. The word count results are displayed in a table format:

fetch	2
few	4
few,	1
fey	1
fie	1
fie,	2
fie:	1
fierie	1
fiery	2
fifty	1
fight	1
fight?	1
fighting	1
fighting,	1
figure	1
figure,	2
figure2	1

The interface also shows navigation buttons for 'First Block', 'Previous Block', 'Next Block', and 'Last Block'. The viewing range is set to bytes 45057 to 49152 of a total 79652 bytes (4096 B block size).

Literature

1. Jerry Zhao, Jelena Pjesivac-Grbovic. 2009.
MapReduce: The programming model and practice, SIGMETRICS
URL : <http://research.google.com/archive/papers/mapreduce-sigmetrics09-tutorial.pdf>
2. Jeffrey Dean, Sanjay Ghemawat. 2004.
MapReduce: simplified data processing on large clusters. In *Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation - Volume 6 (OSDI'04)*, Vol. 6. USENIX Association, Berkeley, CA, USA, 10-10
3. Kazunori Sato. 2012
An Inside Look at Google BigQuery, White paper,
URL : <https://cloud.google.com/files/BigQueryTechnicalWP.pdf>
4. Cloudera Inc.
Example: WordCount v1.0
URL : http://www.cloudera.com/content/cloudera-content/cloudera-docs/HadoopTutorial/CDH4/Hadoop-Tutorial/ht_wordcount1_source.html